Fall 2014

Energy-efficient information inference in wireless sensor networks based on graphical modeling

Wei Zhao
Purdue University

Graduate School Form 30
(Updated 11/20/2014)

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Wei Zhao

Entitled
ENERGY-EFFICIENT INFORMATION INFERENCE IN WIRELESS SENSOR NETWORKS BASED ON GRAPHICAL MODELING

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

Yao Liang

David K.Y. Yau

Arjan Durresi

Xukai Zou

Luo Si

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification/Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

Yao Liang

Approved by Major Professor(s): _____

_____

Approved by: Sunil Prabhakar / William J Gorman 11/30/2014

Head of the Department Graduate Program Date

ENERGY-EFFICIENT INFORMATION INFERENCE IN WIRELESS SENSOR NETWORKS BASED ON GRAPHICAL MODELING

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Wei Zhao

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

December 2014

Purdue University

West Lafayette, Indiana

ACKNOWLEDGMENTS

I would like to thank my advisor, Professor Yao Liang, who stays with me facing rough research phrases and provides expert ideas and guidance. I would also like to thank Dr.Michael G. Ross and Dr.William T. Freeman who selflessly provide very helpful insights based on their probabilistic modeling implementation experience. Some new updates of the thesis were inspired by the questions and ideas from my advisor committee, so I would like to thank all other members (in alphabetic order): Professor Arjan Durresi, Professor Luo Si, Professor David K.Y Yau and Professor Xukai Zou.

TABLE OF CONTENTS

Page

LIST OF TABLES v

LIST OF FIGURES vi

ABSTRACT ix

1 INTRODUCTION 1
1.1 Objectives 1
1.2 Kernel-Based Learning 4
1.3 Optimized Data-Graph 5
1.4 Multi-Resolution Inference 6

2 THEORY BACKGROUND 9
2.1 Undirected Graphical Model 9
2.2 Belief Propagation 12
2.3 Iterative Proportional Fitting 13
2.4 Basic Model Building and Application 16
2.4.1 Basic Model Building and Future Improvement 16
2.4.2 Basic Model Application: Estimation 19

3 KERNEL BASED LEARNING 23
3.1 1D Kernel 23
3.1.1 1D Kernel Methodology 23
3.1.2 Simulation and Analysis 28
3.2 2D Kernel 31
3.2.1 2D Kernel Methodology 31
3.2.2 2D Kernel Simulation-Data 37
3.2.3 2D Kernel Simulation-Modeling 38
3.2.4 Simulation and Analysis 40

4 INFORMATION GRAPH 50
4.1 Methodology 50
4.2 Simulation and Analysis 58
4.2.1 Simulation Data and Setup 61
4.2.2 Simulation with Indoor WSN Data 63
4.2.3 Simulation with Outdoor WSN Data 72

5 MULTI-RESOLUTION INFERENCE 76
5.1 Wavelet Based Belief Propagation 76

Page
5.1.1 Methodology 76
5.1.2 Simulation and Analysis 80
5.2 Multi-Resolution Inference: Based on Data Graph 85

6 SUMMARY AND FUTURE WORKS 93

REFERENCES 99

VITA 105

LIST OF TABLES

Table Page

3.1 Expansion of kernel-based IPF in table 27

5.1 Error severity distribution (dry day; 20 trials) 85

5.2 Error severity distribution (wet day; 20 trials) 85

LIST OF FIGURES

Figure		Page
2.1	An illustration of conditional independent relationship encoded in an MRF.	11
2.2	Demonstration of LBP process.	13
2.3	Flow chart of LBP on basic model.	17
2.4	Topology of the Communication Graph (CG) for IntelLab network.	20
2.5	Performance comparison between Avep-CG and LBP-CG.	21
3.1	Illustration of kernel-based IPF.	25
3.2	Expansion of kernel-based IPF.	26
3.3	Flowchart of simulation of estimation with kernel based learning.	28
3.4	Network topology of the WSN for distributed statistical inference.	29
3.5	Comparison on the estimation performance with different bandwidth options (Accuracy Rate).	31
3.6	Comparison on the estimation performance with different bandwidth options (MAE).	32
3.7	Comparison on the estimation performance between KIPF (h=0.4) and IPF with 60 training samples.	33
3.8	Performance of KIPF (h=0.4) with different number of training datasets.	33
3.9	Performance comparison of KIPF (h=0.4, nTrain=40) with IPFs (nTrain=80, 60, 40).	34
3.10	2D kernel density estimation on data distribution.	36
3.11	2D kernel density dstimation on probability distribution.	37
3.12	Spatial distributions of training data.	39
3.13	Spatial distributions of two test data sets.	39
3.14	LBP realization on pairwise MRF model for sensed data grid.	41
3.15	Estimation performance in terms of accuracy rate with the standard model, 1D and 2D kernel models (Dry, 20 trials).	42

Figure Page

3.16 Estimation performance in terms of MAE with the standard model, 1D and 2D kernel models (Dry, 20 trials). 43

3.17 Estimation performance in terms of accuracy rate with the standard model, 1D and 2D kernel models (Wet, 20 trials). 44

3.18 Estimation performance in terms of MAE with the standard model, 1D and 2D kernel models (Wet, 20 trials). 45

3.19 Robustness analysis based on estimation performance with 2D kernel models (Dry, Accuracy Rate, Increase by 0.1). 46

3.20 Robustness Analysis based on estimation performance with 2D kernel models (Dry, MAE, Increase by 0.1). 46

3.21 Robustness analysis based on estimation performance with 2D kernel models (Dry, Accuracy Rate, Change by 0.5). 47

3.22 Robustness analysis based on estimation performance with 2D kernel models (Dry, MAE, Change by 0.5). 47

3.23 Robustness analysis based on estimation performance with 2D kernel models (Wet, Accuracy Rate). 48

3.24 Robustness analysis based on estimation performance with 2D kernel models (Wet, MAE). 48

3.25 Robustness analysis based on estimation performance with 2D kernel models (Dry, Accuracy Rate, Inappropriate range). 49

3.26 Robustness analysis based on estimation performance with 2D kernel models (Dry, MAE, Inappropriate range). 49

4.1 Inference in DG vs inference in CG (with network topologies). 51

4.2 An outline of our approach to information structure optimization for distributed in-network inference in WSNs. 53

4.3 Flowchart of simulation of estimation with DG. 59

4.4 MAE comparison between LBP based on Avep and uniform priors. . . 60

4.5 Illustration of IntelLab topology. 62

4.6 Illustration of Redwood topology. 62

4.7 Comparison of testing performance of CG and DG with MAE. 65

4.8 Validation performance (training data size: 80). 65

4.9 Comparison between CG and DG topologies. 67

Figure Page

4.10 Comparison of testing performance of CG and DG on MAE. 68
4.11 Topology reduced based on geographical distance 69
4.12 Comparison of estimation accuracy. 70
4.13 Transmission and reception counts. 71
4.14 Topology of CG for the Redwood WSN. 73
4.15 Topology of DG for the Redwood WSN (EdgeNum28). 73
4.16 Validation performance different lambda/number of edges. 74
4.17 Estimation performance comparison of CG and DG (EdgeNum28). . . 75
4.18 Energy analysis based on message count. 75
5.1 Decomposition and reconstruction. 78
5.2 W-LBP process. 79
5.3 Flowchart of simulation of estimation with W-LBP. 81
5.4 Accuracy comparison between W-LBP and LBP (dry). 83
5.5 Accuracy comparison between W-LBP and LBP (wet). 84
5.6 Geographical distribution of errors of LBP. The black dots represent the positions of missing readings, and, on top of them, green, blue or red square indicates error level one, two or three. All the rest positions are the sites correctly estimated. 84
5.7 Flowchart of simulation with Data Graph based Multi-Resolution Inference. 87
5.8 Demonstration of DG based Multi-Resolution Inference. 88
5.9 Topology of DG for Multi-Resolution Inference (EdgeNum of DG=77). 89
5.10 Performance comparison (IntelLab: EdgeNum of DG=77). 90
5.11 Energy efficiency comparison to CG. 90
5.12 Energy efficiency comparison to DG. 91
5.13 Topology of DG for Multi-Resolution Inference (EdgeNum of DG=22). 91
5.14 Performance comparison (Redwood: EdgeNum of DG=22). 92
5.15 Energy efficiency analysis for outdoor sensor network. 92

ABSTRACT

Zhao, Wei Ph.D., Purdue University, December 2014. Energy-efficient Information Inference in Wireless Sensor Networks Based on Graphical Modeling. Major Professor: Yao Liang.

This dissertation proposes a systematic approach, based on a probabilistic graphical model, to infer missing observations in wireless sensor networks (WSNs) for sustaining environmental monitoring. This enables us to effectively address two critical challenges in WSNs: (1) energy-efficient data gathering through planned communication disruptions resulting from energy-saving sleep cycles, and (2) sensor-node failure tolerance in harsh environments. In our approach, we develop a pairwise Markov Random Field (MRF) to model the spatial correlations in a sensor network. Our MRF model is first constructed through automatic learning from historical sensed data, by using Iterative Proportional Fitting (IPF). When the MRF model is constructed, Loopy Belief Propagation (LBP) is then employed to perform information inference to estimate the missing data given incomplete network observations. The proposed approach is then improved in terms of energy-efficiency and robustness from three aspects: model building, inference and parameter learning. The model and methods are empirically evaluated using multiple real-world sensor network data sets. The results demonstrate the merits of our proposed approaches.

1 INTRODUCTION

1.1 Objectives

With the development of sensor module design and miniaturization, small micro-electrical-mechanical sensor (MEMS) modules have become cheaper and yet more powerful. These sensing modules are designed to combine sensing, processing, data storage and communication capabilities. It is possible today to deploy wireless sensor networks (WSNs) consisting of thousands of tiny sensor nodes to work jointly and collect various sensed data for scientific discoveries and engineering applications which have never been possible before, including geographical surveillance, environmental monitoring, ecological studies, hydrological studies, climate recording, and engineering construction monitoring ([1], [17], [31]).

In this work, we are concerned with sustaining environmental monitoring wireless sensor networks usually deployed in harsh or even hostile environments such as forests, mountainous areas and oceans, where each sensor node has severe battery power limitation and the replacement of battery is usually impossible. Sustaining monitoring is different from other categories of WSNs aimed for event detection and object tracking, in which sensor nodes transceivers can be in sleep mode most of the time when no significant events are detected. In contrast, sustaining monitoring WSNs require continuous gathering and recording of observations on the deployed spots for sophisticated studies offline in order to understand/discover the fundamental nature and laws of the physical processes in question, such as environmental, ecological, hydrological, and climate studies.

Two critical challenges in environmental WSNs that we face today are: (1) energy-efficient data collections to maximize the lifetime of the WSNs in real-world operation and simultaneously provide high quality of observation data, and (2) robustness with

respect to unexpected communication disruptions due to sensor nodes failures. Current energy-efficient approaches in WSNs include energy-aware routing for ad hoc sensor networks([33], [37]), energy-efficient medium access control (MAC) protocols and intelligent resource allocation ([47], [69]), and adaptive sampling([4], [58]). One important direction is to exploit spatial and temporal correlations in the observations from dense WSNs. A large body of research exists in recent years. For instance, a theoretical framework is developed to model the spatial and temporal correlations in WSNs and a correlation-based MAC protocol is designed for object tracking([63], [64]); in [16], a HMRF (hidden Markov Random Field) model is presented for distributed estimation from the noisy measurements applied to event-region detection. For environmental monitoring WSNs, most of existing work includes source coding, data-centric routing, and sourcing coding coupling with routing (e.g., [10], [13], [23]). Distributed source coding approach conducts compression of multiple correlated sensors outputs and jointly decodes them at the sink. This approach though requires tracking the data and sending a unicast message to each sensor node about its coding parameters once a while; it is also difficult to obtain a joint probability density function in a WSN. Data centric routing seeks to reduce data sets by in-network processing and aggregation. On the other hand, while a few methods exist to address the robustness issue in terms of sensor node failures in WSNs for event/target detection ([11], [42]), little work on fault-tolerance has been reported regarding node failures in WSNs for sustaining environmental monitoring. In this work, we present a novel systematic approach to address the both challenges of energy-efficient data collections and fault-tolerance in sustaining environmental monitoring WSNs. We focus on data collections in WSNs where all sensor observations along time have to be gathered and stored at the sink for offline scientific discovery and analysis, as opposed to query answering applications [15]. Our approach is based on probabilistic graphical model [74]. We adopt Markov Random Fields (MRFs), undirected graphical models, to model the spatial correlations in environmental sensor networks, inspired by recent researches on successfully applying MRFs in computer vision ([18], [21], [22]). Unlike

the work in [14] assumeing the correlation matrix of its graphical model is given, our approach, through automatic learning to establish the correlation model, consists of two major phases. First, an MRF model is constructed through automatic learning from historical sensed data. Then, when the MRF model is constructed, only partial sensor network observation is needed in data collection, and the missing observations are estimated through information inference using the constructed MRF model at the sink. Thus, our approach provides a unified framework not only to deal with the robustness issue of sensor node failures, but also to address energy-efficiency for data collections in WSNs, as sensor nodes can be put into sleep mode periodically to significantly reduce the energy consumptions and the missing samples of sleeping nodes are inferred from other active nodes observations at the sink. In contrast with the work based on simple Bayesian inference in [32], all parameters of our model will be learned through automatic learning, without any assumption for prior distribution of data, nor any need for simulations of parameters. Furthermore, our approach is general, regardless of any concrete MAC layers and/or routing algorithms being used. In other words, our approach can readily work with any existing WSN MAC layers, routing algorithms and data gathering communication protocols to achieve dramatic energy savings for data collections. In contrast, most existing approaches of exploiting spatial correlations in WSNs require to developing special MAC mechanisms or routing schemes or communication coding algorithms in order to reduce power consumptions. Moreover, our proposed approach can also be used jointly with other existing approaches for energy conservation in WSNs to achieve even more energy conservation, including energy-efficient routing, MAC protocols, scheduling and error control, ([33], [37], [69]), and data compression ([13], [34]).

To design our systematic information inference approach for missing observation estimation in WSNs, not only we want to achieve a satisfying performance in term of estimation accuracy, the on-board energy restriction of sensor nodes, one of the most important concerns of data collection mechanism in a WSN, introduces more challenges that we must consider to make our methods fit the requirements of real-world

WSN applications. Correspondingly, on top of the basic inference model introdued in Section 2.4.1, we designed different methods for energy-efficiency purpose for information inference and learning process respectively, which are the main contribution of the thesis. The general ideas of those methods are introduced in the following sections.

1.2 Kernel-Based Learning

When information inference can raise problems in energy-sensitive applications of WSNs, we also examined the energy efficiency requirement of learning process, to make it feasible in a real WSN application. Once an MRF is constructed for a target WSN, distributed statistical inference in the WSN can be then cast as a problem of computing either the marginalization or the maximum a posteriori (MAP) configuration in the MRF, which can be solved by various efficient message-passing algorithms. However, one important step before information inference will focus on how to effectively and efficiently learn parameters of pair-wise MRF model for a deployed WSN. While data-driven approach is usually preferred, a critical challenge is the rareness of available training data. It is usually expensive or even impossible to collect a large amount of training samples in a deployed wireless sensor network, due to the constrained resource (e.g., motes power) and/or time urgency of the task. One essential question one may ask is how to learn the graphical model parameters of a deployed sensor network using as few training data as possible without affecting the constructed graphical models effectiveness. In view that Iterative Proportional Fitting (IPF) procedure is an appropriate and preferred method for learning MRF parameters due to its computational speed and numerical stability, we propose a kernel-based approach for MRF parameters learning for a modeled sensor network. Kernel methods have been fruitful for statistical classification and regression in recent years. In WSN related fields, kernel-based learning has been presented for sensor network and wireless local area network localization (e.g., [53], [40]). With our kernel-

based learning method, we show that it is possible to substantially reduce the number of training samples needed for MRF parameter learning by our proposed kernel-based IPF approach compared to the original IPF procedure with comparable model inference performance. We demonstrate our approach by rigorous simulations using real-world data from two real wireless sensor networks (an indoor network and an outdoor network), in which the standard IPF procedure is employed as a baseline and the results are carefully analyzed.

1.3 Optimized Data-Graph

While we consider belief propagation in Chapter 3, we assume the communication topology of the targeted WSN is casted directly as the topology for inference. However, to make the belief propagation cooperate the properties of a WSN and be more energy efficient we should consider producing a network structure tailored specially for information inference, which we referred as Data Graph, as opposed to the Communication Graph mapped directly from communication connection of a sensor network.

For the research of adoption of the BP-based algorithms as the basis for WSN communication and distributed inference is very promising, most recent work focuses either on the improvement of standard BP algorithms, such as reweighted belief propagation algorithms ([8], [55]) to address the convergence problem of loopy BP; on the improvement of energy efficiency and scalability of BP-based approaches (e.g., [36], [73]); or on methods for mapping BP into a practical WSN with constraints [2]. Although, as noted in [7], the information structure of the inference problem is as critical as the communication structure of the problem, little research has been reported regarding how to systematically build an appropriate WSN information structure. The lack of research in this emerging area motivates our work of investigating an optimized information structure especially achieved for information inference. We refer to the information structure of a WSN as Data Graph (DG). In contrast, the com-

munication topology of a WSN is referred to as Communication Graph (CG). The current practice of statistical inference in WSN is mainly carried out based on WSN CG, regardless of the inference method used. In other words, the DG of a WSN application is implicitly defined by the WSNs CG. In some specific applications described in ([7], [36]), their graphical model formation is obtained by adding edges to the CG. The resulting information structure would be an augmented DG from the CG of a WSN. In general, when a graphical model is built on top of CG, it is important to guarantee that message-passing on the graphical model can be implemented with only one-hop communication in the CG, in which no-routing for inference messages is required (this property is referred to as the no-routing property [55]). Obviously, an augmented information structure does not satisfy the no-routing property, making message-passing on the DG of a WSN inefficient in practice.

To address this problem, we propose a general data-driven approach to systematically obtain a more effective information structure given any communication structure of a sensor network, upon which BP-based distributed inference can be more effectively and efficiently performed. Our approach is based on graphical model optimization, and thus is theoretically sound and rigorous. To the best of our knowledge, our work described here is the first to apply graphical model optimization to the estimation in a WSN.

1.4 Multi-Resolution Inference

With the advances of current technologies, it is possible to deploy WSNs consisting of thousands of tiny sensor nodes for various monitoring tasks. In such applications, sensor nodes need to process the information collected from each node jointly for information fusion, and for handling sensor/node failures resulting in contaminated or missing readings. Due to its compact representation, distributed propagation and robustness property, Loopy Belief Propagation (LBP) has been proved to be theoretically appropriate and naturally suitable for handling uncertainties on-line in WSNs

through well organized belief message transmissions and simple local belief updates (e.g., [12], [9]). However, as individual sensor nodes only have limited battery power, the energy consumption resulting from frequent belief message exchanges in LBP tends to be a serious problem. To handle this problem, we present multi-resolution inference method to reduce the size of belief message when retaining the most important information for estimation. It includes two versions: the basic version based on only wavelet transformation of belief messages and the extended version combining structure optimization, in which only messages outside Data Graph will be processed by wavelet transformation. The proposed multi-resolution inference can significantly reduce the transmission of belief messages in the traditional LBP through wavelet transformation on belief messages, making LBP-based in-network inference more energy efficient and thus more suitable for real-world WSN applications. It is desirable for the multi-resolution inference not only to achieve significant energy conservation but also to minimize any possible degradation of estimation performance at the same time.

A few studies to improve LBPs energy efficiency in WSNs exist. One recent idea is to take advantage of multicasting in WSNs to multicast identical one-to-multi-target messages from a sensor node to its neighborhood, instead of distinct ones required in traditional LBP [5]. Another attempt is to schedule message passing based on whether an individual node has sufficient new information to warrant the transmission of a new message, to avoid the energy cost of transmission of uninformative messages [7]. Yet another promising approach is called Nonparametric Belief Propagation (NBP) (e.g., [36], [35]), which reduces the communication volume of individual messages through sample-based approximation of each message. Unlike NBP targeted for non-Gaussian continuous random variables, the proposed multi-resolution inference focuses on discrete random variable cases. While NBP and multi-resolution inference are fundamentally different, both approaches share two basic ideas: (1) to approximate messages with less bits to conserve energy during message communication, and (2) to perform such approximation with only local information without

causing extra communication. Our approach is applicable to various WSN applications for which LBP inference is suitable, including target tracking, hypothesis testing, self-calibration and network clustering.

Based on the experiment with real-world WSN data, multi-resolution inference can significantly reduces the communication volume during distributed belief inference, with very minimal degradation of estimation performance. The proposed multi-resolution inference thus could become a better and more realistic communication basis to support distributed inference in WSNs for various applications where energy saving and performance robustness are crucial due to the severe energy limitation of sensor nodes. We demonstrate our approach through in-network estimation application. Haar wavelet was chosen due to its simplicity to implement in sensor node. Although only one level of wavelet decomposition is illustrated in our empirical study, multilevel decomposition on data graphs with different edges resulted from different values of λ can be adopted to achieve more substantial energy conservation. Therefore, the proposed multi-resolution inference provides full flexibility to tradeoff inference performance with energy efficiency and opens up a new design and operational space to optimally match the specific objectives of WSNs under resource constraints. Also, due to the nature of localized communications of distributed belief inference in WSNs, the multi-resolution inference inherits the scalability of the original LBP.

2 THEORY BACKGROUND

As the platform of our probabilistic inference method, undirected graphical model or a MRF is applied to model a WSN to address both unexpected and planned communication disruptions during WSN data collections for sustaining environmental monitoring. The introduction of MRFs can be a natural starting point to understand the whole approach by setting up a framework for inference algorithm, like LBP, and parameter learning algorithm, IPF, and the algorithms aiming to improve these two processes.

2.1 Undirected Graphical Model

An MRF is a set of n random variables indexed over the vertices, or sites in an ordered lattice. In an MRF, each node is independent with other nodes conditioned on its neighboring system [41]. For first order MRFs, the neighbor system for each node only contains those with direct connections to it. The MRF variables are not independent, but are mutually coupled; a key property of MRFs is that the distribution of the random variable associated with a site n given the values associated with the sites in the neighborhood of n, is independent of the rest of the sites in the MRF. This can be formalized as

$$P(X_n = x_n | X_t = x_t, \forall t \neq n) = P(X_n = x_n | X_t = x_t, \forall t \in n) \tag{2.1}$$

where x_n denotes the random variable of site n and N_n is the set of random variables associated with the sites in the neighborhood of site n. MRFs are in some way similar to Bayesian Networks, but their joint distributions cannot be computed by multiplication of local conditional probability functions. Therefore, this conditional independent relationship cannot contribute much to the computation in an MRF

model. To represent an MRF model in a factorized formulation, we need to borrow some ideas from Gibbs Random Fields (GRFs). A GRF is a random field that has Gibbs distribution

$$P(W) = \frac{1}{Z} exp(-\frac{1}{T} U(W)) \tag{2.2}$$

$$Z = \sum_{W \in \Omega} exp(-\frac{1}{T} U(W)) \tag{2.3}$$

where Z is a partition function, T is a constant called Temperature, W is a concrete configuration over the involved variables; Ω is the set of all possible configurations; and $U(W)$ is the energy function. $U(W) = \sum_{c \in C} V_c(W)$ is a sum of clique potentials over all cliques C . In a GRF, prior is factored over clique c :

$$P(x) \propto \exp(-\sum_{c \in C} V_c(x_c)) \tag{2.4}$$

which is often written as

$$P(x) \propto \prod_{c \in C} \psi_c(x_c) \tag{2.5}$$

MRF is characterized by its local property (markovian), while GRF is characterized by its global property. Hammersley-Clifford theorem indicates that an MRF is equivalent to a GRF if it correctly captures the conditional independencies of the distribution ([24], [60]). More precisely, if distribution P is everywhere non-negative, and G is an I-map of P, then can be factorize over in the form

$$P(x) = \frac{1}{Z} \prod_{c \in C} \psi_c(x_c) \tag{2.6}$$

where c are cliques of the graph. Cliques are fully connected neighborhoods. The potential function ψ_c need not sum to one, so the global normalization constant Z is needed:

$$Z = \sum_{x} \prod_{c \in C} \psi_c(x_c) \tag{2.7}$$

If we represent the set of conditional independencies for distribution using $P(x)$ and we can get the conditional independent relationships encoded with a MRF, we say

there exist an I-map relationship if Eq. 2.8 holds. For example, given a graph in Figure 2.1, we can get the global Markov independencies from the topology as

$$I(G) = \{(X_1, X_5 \perp X_3 | X_2, X_4), \cdots\} \tag{2.8}$$

These independent relationships should be included by the true independent properties of the joint probability distribution to represent it as a GRF. Assuming an MRF model is parameterized by θ, we then represent its joint probability as

$$P(x|\theta) = \frac{1}{Z(\theta)} \prod_{c \in C} \psi_c(x_c|\theta_c) \tag{2.9}$$

and

$$Z(\theta) = \sum_{x} \prod_{c \in C} \psi_c(x_c|\theta_c) \tag{2.10}$$

To perform information inference in an MRF model, certain prior knowledge is needed. The prior knowledge comes from two aspects: the correlations among neighbors, and the current partial observations from the neighborhood of node(s) in question. A key is to effectively and efficiently obtain the correlations in a neighborhood. Desirably, such correlations for the MRF model could be built through automatic learning from historical observations. The information inference is to fuse all the above prior knowledge to infer the marginal probability distribution on the node(s) of interest.

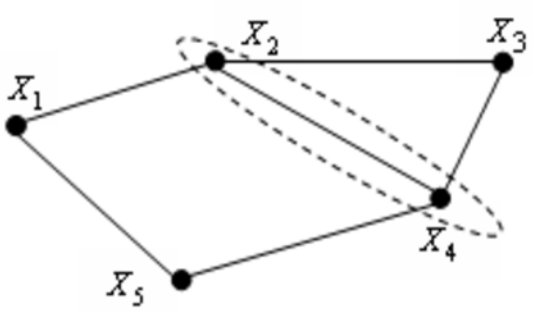


Figure 2.1. An illustration of conditional independent relationship encoded in an MRF.

2.2 Belief Propagation

Pearl originally introduced the Belief Propagation (BP) algorithm (the polytree algorithm) for the solution of inference in directed graphical model which is singly-connected [49]. It is a decentralized iterative algorithm that operates by message transmission among nearby nodes in a probabilistic graphical model. Loopy Belief Propagation is an approximation scheme of BP applied to loopy graphical model in an iterative manner. While the BP algorithm is known to converge to the correct solution for singly connected networks, for loopy networks the beliefs may not converge and even if BP dose converge it may not converge to the correct solution. However, several groups have reported excellent empirical results by using LBP, and the convergence and accuracy properties of this algorithm are discussed ([46], [70]). For instance, Turbo-Codes error-correcting coding algorithm is a great example of BP application to a loopy graphical model, which has been considered as the most exciting and potentially important development in coding theory in many years [44]. In Figure 2.2, we briefly illustrate how LBP works in pairwise MRFs. In LBP, a variable message m_{ij} is introduced, and each node i sends a message m_{ij} to each of its neighbors j. A message m_{ij}, a vector of the same dimensionality as x_j, is to inform node j which values it thinks are most likely for x_j; and updates its belief (i.e., the node marginal), based on the messages it receives from its neighbors, as follows:

$$m_{ij}(x_j) \leftarrow \sum_{x_i} \psi_i(x_i)\psi_{ij}(x_i, x_j) \prod_{k \in N(i)\setminus j} m_{ki}(x_i) \tag{2.11}$$

$$b_i(x_i) = K\psi_i(x_i) \prod_{j \in N(i)} m_{ji}(x_i) \tag{2.12}$$

where K is normalization constant and $N(i)$ denotes the neighbors of node i. Note the principle behind this message transmission scheme is that, any node i has to compute m_{ij} based on the belief messages coming from its directly connected neighbors except for node j it is going to send the update message to.

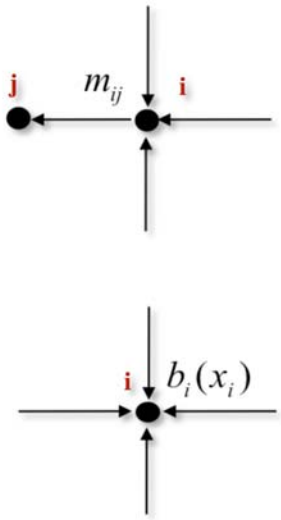


Figure 2.2. Demonstration of LBP process.

2.3 Iterative Proportional Fitting

It is desirable to learn the pairwise MRF model of WSN through automatic learning. To do so, an appropriate learning algorithm is critical. We believe that Iterative Proportional Fitting (IPF) algorithm is a good fit for this purpose. IPF is a mathematical scaling procedure that can iteratively adjust each element of the original two-dimensional table to fit constraints assigned on the row and column totals, ([6], [19], [20], [39]). If we consider this table as a discrete joint probability distribution, by performing IPF, we can get its maximum likelihood estimation. That is, IPF allows us to make an ML estimation of a joint distribution from observations of various marginal distributions. If empirical marginal probability distributions are available, then we can get the estimated discrete joint probability distribution at the ML point using IPF:

$$\hat{n}(x_1,x_2)^{(1)} = \hat{n}(x_1,x_2)^{(0)} \times (n_1(x_1)/\hat{n}_1(x_1)^{(0)}) \tag{2.13}$$

$$\hat{n}(x_1, x_2)^{(2)} = \hat{n}(x_{1,}x_2)^{(1)} \times (n_2(x_2)/\hat{n}_2(x_2)^{(1)}) \tag{2.14}$$

where $\hat{n}$ denotes the expected value of the elements of joint probability. It is worth mentioning that Eqs. 2.13 & 2.14 do not imply any order requirement between two iterations, where the order can be arbitrary in theory. We can exam the procedure of IPF with a simple two-dimensional example. Given initial values for the binary variable x_1, x_2 as:

	$x_2 = 0$	$x_2 = 1$	$n_1(x_1)$
$x_1 = 0$	1	1	40
$x_0 = 1$	1	1	60
$n_2(x_2)$	60	40	

In the first step, we update the value of each cell according to the marginal $n_1(x_1)$ by Eq. 2.13 & 2.14. That is for each cell, when $x_1 = 0$, we have $x_{i,j} = 40 \times 0.5 = 20$ and $x_{i,j} = 60 \times 0.5 = 30$ when $x_1 = 1$. In the same way, we update according to marginal $n_2(x_2)$ and get $x_{i,j} = 24$ and $x_{i,j} = 16$. The final estimation result is

	$x_2 = 0$	$x_2 = 1$	$n_1(x_1)$
$x_1 = 0$	24	16	40
$x_0 = 1$	36	24	60
$n_2(x_2)$	60	40	

As a natural extension, we can further derive the IPF update rule for a pairwise MRF from this example:

$$\psi_c^{t+1}(x_c) = \psi_c^t \frac{\tilde{P}(x_c)}{P^t(x_c)} \tag{2.15}$$

where superscript t denotes the round of the iterations, $P^t(x_c)$ the estimated marginal distributions, $\tilde{P}(x_c)$ the empirical marginal distributions and ψ_c^t the estimated poten-

tials at iteration t. In such an iterative process, IPF will scan through all the potentials and make local changes to increase the probability of the overall assignment. Considering an MRF with discrete space, the MRF configuration of the state space depends on several factors including the dynamic range of physical variables being monitored, the accuracy of sensor type being used, and the requirements of applications. Assume a finite number of possible joint settings, for a particular dataset with N data samples, we can count the number of times any joint configuration has been observed:

$$n(x) = \sum_n \delta(x, x^n) \tag{2.16}$$

We can also count the number of times a clique configuration appears:

$$n(x_c) = \sum_n \delta(x_c, x_c^n) \tag{2.17}$$

In terms of the counts, the log-likelihood is given by

$$P(D|\theta) = \prod_n \prod_x P(x|\theta)^{\delta(x,x^n)} \tag{2.18}$$

$$\log P(D|\theta) = \sum_n \sum_x \delta(x, x^n) \log P(x|\theta) \tag{2.19}$$

We can see from Eq. 2.18, the clique counts are the sufficient statistics for our MRF model, so we can use IPF for the ML estimation. To obtain the ML estimation, we calculate the derivative of the log likelihood with respect to the value of one clique potential and set this derivative to zero, trying to find the optimal parameters:

$$\tilde{P}(x_c) = \frac{n(x_c)}{N} = P_{ML}(x_c) \tag{2.20}$$

For each iteration, belief inference is required to compute marginal. To solve it directly will be hard because it appears on both sides of this implicit nonlinear equation. The idea of IPF is to hold fixed on the right hand side and solve for it on the left hand side. That is, we cycle through all cliques, and iterate Eq. 2.15. In other words, at the maximum likelihood setting of the parameters, for each pairwise clique, the

model marginal distribution must be equal to the observed marginal distribution (normalized empirical counts). The IPF algorithm iteratively enforces individual marginal constraint to each potential. By iterating over all such constraints IPF monotonically converges to a unique solution when one solution exists. For our WSN task, all the nodes are discrete and each potential is represented with a table (each entry of this table corresponds one state combination of all the nodes involved in the clique, to which the potential attached). It is easy to see that we can get the empirical marginal distributions from the training datasets, but it is hard to get the expected model marginal distributions since there is no closed form to use. If one wants to infer it from the model using LBP, it will dramatically increase the complexity and computation of IPF. A more sophisticated solution we have is based on the discovery in [65] which confirmed the existence of a fixed point, in the following Eq. 2.21, of IPF in the process of belief propagation. Accordingly, algorithms in this class can be formulated as a sequence of reparameterization updates, each of which entails refactorizing a portion of the distribution corresponding to an acyclic subgraph. We therefore utilize the fixed points to simplify the process of IPF. During the reparameterization, a fixed point exists for each edge [54],

$$\psi_{st}(x_s, x_t) = \frac{P_{st}(x_s, x_t)}{P_s(x_s)P_t(x_t)} \tag{2.21}$$

We apply this fixed point to get the parameters of the pairwise MRF.

2.4 Basic Model Building and Application

2.4.1 Basic Model Building and Future Improvement

We need to first understand the target problem for building a well functional model. As mentioned in Section1.1, it is necessary to develop a model that can effectively collect data, in term of data quality and energy efficiency, and maximize the lifespan of a sensor network, especially the one containing large number of sensors. This requirement call for a model that is robust, able to recover missing observation

and distributes the load to all sensors without draining out particular group of sensors, which can significantly shorten the life of the whole sensor network. To meet those requirements, a suitable model based on MRFs/Undirected Graphical Model and Belief Inference is built ([72], [74]). In this model, each sensor node is mapped to a node in a MRF and the correlation relationship in a neighborhood is encoded as potential function on each edge. The Belief Inference is performed in such model, fusing two parts of information, partial observation and correlation information, for an optimal and distributed estimation of the missing observations. The main procedure of the experiment is illustrated in Figure 2.3 for a clearer image of basic probabilistic inference.

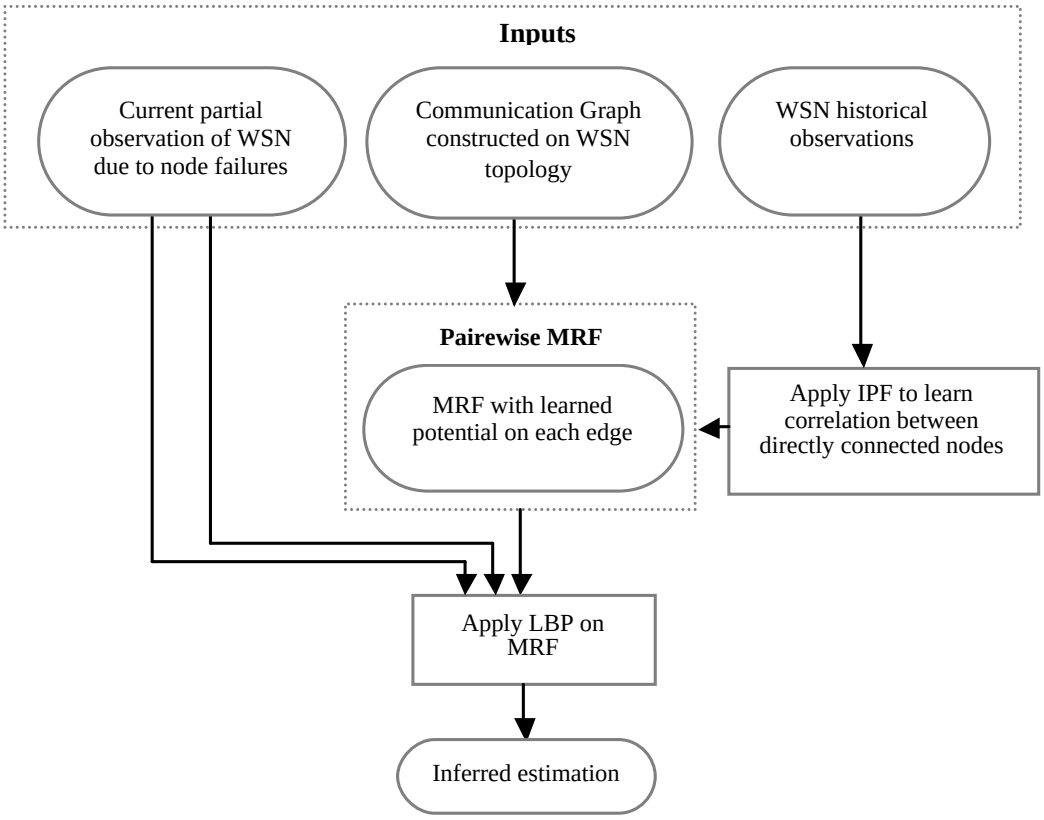


Figure 2.3. Flow chart of LBP on basic model.

To evaluate the performance of the proposed model, we also establish another model for comparison which dose not extract probabilistic spatial information within neighborhood. To estimate unobserved nodes, an intuitive method is used, which computes the average of observed neighbors of an unobserved node as estimation. This simple estimation method however cannot handle the situation whenever a missing node has none of observed neighbors. To overcome this weakness, we introduce an iterative average procedure Avep() as follows. Denote the set of all currently unknown nodes/points in the MRF model as M. Denote the set of all observed (or estimated) neighbors of node i as $N(i)$, and the belief distribution of node i as $BEL(i)$, respectively. Then the Avep() procedure can be given as follows:

Pseudocode: Avep()

```
begin
  while (M ≠ ∅)
    Get next node i ∈ M;
    if (N(i) ≠ ∅)
        BEL(i) ←Ave( N(i));
        M ← M − {i};
    end if
  end while
end
```

In our empirical study, to evaluate the performance of our approach, we use Avep() as the performance baseline to estimate unobserved data points. In addition, the local beliefs from Avep() will be used as the priors of the missing observations for the Belief Inference in our MRF model. The target estimation method is LBP inference on the CG, with partial observation as priors as shown in Figure 2.3.

The estimation performance model we use is mean absolute error (MAE) suggested by [67], which has been shown to be a natural measure for average model error that

has better reliability than root-mean-square error (RMSE) with the variability of error distribution [67]

Until now, a basic model for data collection has been established to handle missing data in a sensor network. However, this is just the first step, the framework for further development for handling various important challenges: 1) energy efficiency of Belief Propagation; 2) data spareness during parameter training. As for energy efficiency, we cope with this from two aspects: structure optimization and transmission reduction. For data spareness, caused either by lack of data or energy conservation process, we try to extract more information with limited data available for parameter training. Those improvements will be discussed in details in the following chapters.

2.4.2 Basic Model Application: Estimation

We conducted intensive simulations to thoroughly evaluate the proposed inference method, in which real-world WSN data is used. The data is collected from the indoor WSN of Intel Berkeley Research Lab. Our in-network inference application is to estimate missing readings of sensors for the WSN via the distributed inference using the collected data set from that WSN, where some original sensor readings are set aside to evaluate the estimation performance. The distributed in-network inference is performed with LBP. The in-door sensor network of Intel Berkeley Research Lab consists of 54 Mica2Dot motes, operating on TinyOS, spreading over the whole lab, and we select 50 motes with enough temperature readings in this simulation. It is reasonable to assume that the room temperature ranges from 15 to 30 degrees Celsius and thus can be descretized into 15 discrete states with the constant step as 1 degree.

We will use Communication Graph (CG) as our pairwise MRF model for this application, each node represents a mote (with its temperature sensor), and each edge represents a communication link. To build a CG for the Intel Berkeley WSN, we select four nearest neighbors to construct the neighborhood based on [68]. According to the aggregate connectivity statistics provided by the Intel Berkeley Lab, shorter

distances do lead to lower packet dropping rates, justifying our CG model. The number of neighbors chosen is to meet the robustness consideration in [68]. The topology of CG for IntelLab is shown in Figure 2.4

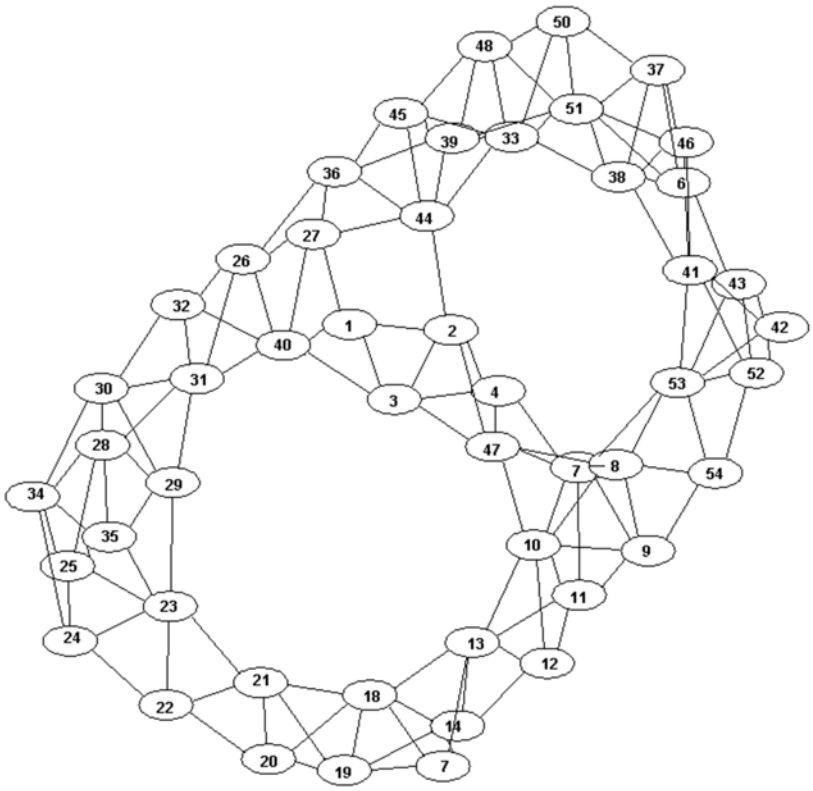


Figure 2.4. Topology of the Communication Graph (CG) for IntelLab network.

When CG is established, we need to decide the correlation information for each link in CG, which will be extracted by IPF learning. The total 80 training sets are available to learn the potential associated to each link and 10 other data sets are reserved for testing in our simulation. For each test case (i.e., test data set), we randomly select a fraction of motes with missing readings to be the estimating targets of the application. We assign Avep on CG (Avep-CG) as priors over all motes of missing readings, and perform the LBP (LBP-CG) inference based on those prior information.

To thoroughly evaluate the estimation performance of those two estimation methods, the number of unobserved motes in our simulation is gradually increased. Estimation by LBP-CG involves partial observations and the correlation information among neighbors, extracted by parameter learning, while the benchmark estimation Avep() uses partial observations alone. Thus, LBP-CG is expected to outperform Avep() as shown in Figure 2.5 where the red and blue color represent the MAE for Avep and LBP respectively.

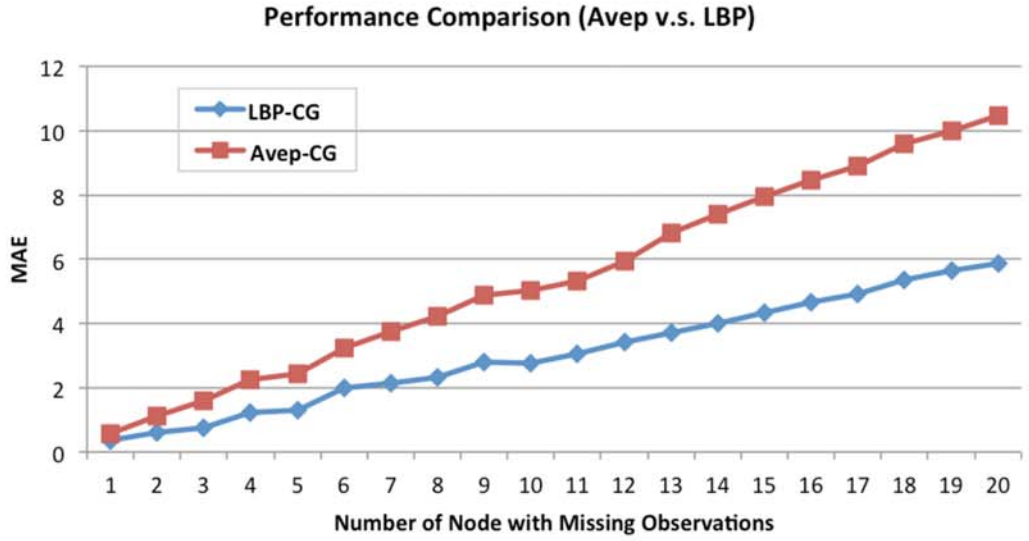


Figure 2.5. Performance comparison between Avep-CG and LBP-CG.

The higher value of MAE indicates more serious model error during the estimation. As we expect, the performance of LBP is always above Avep, and the gap tend to increase with the percentage of missing nodes increasing. Since the configurations for Avep and LBP are the same, there are two observations based on the comparison in Figure 2.5: 1) the extracted correlation information through IPF contributes to the improvement of performance; 2) the LBP inference process fuses this support information from different sources, other than partial observation, to benefit the final estimation performance. Actually, To eliminate the impact caused by the node con-

figuration randomness of missing observations, we average the estimation accuracies over 30 runs for all test cases.

3 KERNEL BASED LEARNING

3.1 1D Kernel

3.1.1 1D Kernel Methodology

The central problem of pairwise potential learning is to get the appropriate distribution estimations. With the common choice of histogram method for estimating distributions, MRF modeling of WSN via IPF learning could be done with sufficiently large amount of training data, which would be expensive or even prohibited in a real WSN application due to the severe resource limitation of WSN for data collection. To address this challenge, we introduce kernel method [75], as opposed to the histogram method, into the IPF learning for MRF modeling, where a probability mass is assigned to a kernel of each observation in training data. Comparing to the traditional histogram method, the kernel method avoids the tricky dependence on the choice of the boundary points of bins, and shows better mean square error rate ([59], [57]). The kernel idea was first raised as a smoothing technique for multinomial cell probability by Good ([25], [26]). The general form of smoothing estimator can be presented as

$$\hat{p}_i = \sum_{j=-\infty}^{j=\infty} K(i,j,h)\hat{p}_j \qquad i,j \in I \tag{3.1}$$

where $K(i,j,k)$ is the weight function or can be considered as kernel, h is the bandwidth parameter and $\hat{p}_j$ denotes the relative frequency of cell j. More research has been done by choosing appropriate weight function, e.g. Wang and Van Ryzin [66] presented a class of estimator with Geometric Kernel, which has rapid drop off features. The smoothing ability of this kernel is limited when the author chooses the bandwidth by truncated the kernel function using MSE criterion. The further de-

velopment in work of [30] formalized smoothing estimators in the similar way by replacing weight function K as W:

$$W(i,j,h) = \frac{K(\frac{i-j}{h})}{s(h)}, \quad h > 1 \quad \text{and} \quad s(h) = \sum_{j=-\infty}^{j=\infty} k(\frac{j}{h}) \tag{3.2}$$

where K can represent any suitable continuous univariate kernel function. Such smoothing function is where we bought idea from, but with two main differences: 1) the mean of kernel function is located by individual measurement, which is fixed on the median of certain cell in traditional smoothing algorithm; 2) the kernel function is applied to joint discrete probabilistic function, instead of univariate function smoothing. These two main differences are discussed in more detailed in later part of this section. In our approach, on node i in an MRF model, the smoothing property is applied to training set $S_k(k = 1 : N)$ by scaled kernel function K_i as

$$K_i^h(x - s_k) = \frac{1}{h_i} K_i(\frac{x - s_k}{h_i}) \tag{3.3}$$

where h_i is the smoothing parameter, the bandwidth, for node i. Gaussian kernel function is usually a popular choice which is also adopted in our approach. The process of building Gaussian kernel function for parameter learning is demonstrated in Figure 3.1.1. In Figure 3.1.1, each black dot indicates one training data point which will form the center of one new Gaussian kernel function and exert independent impact to its adjacent discrete states.

Thus, according to Eq. 3.3, we have kernel function as

$$K_i^h(x, s_k) = \frac{1}{h_i\sqrt{2\pi}} e^{-\frac{(x-s_k)^2}{2h_i^2}} \tag{3.4}$$

and we now get the weighted average following a probability density function over a certain discrete span. For a discrete case, when a certain sample appears in the training dataset, we have

$$\sum_{k}^{S} \frac{1}{h_i} \int_{Rb_k}^{Lb_k} \phi_i(\frac{x - s_k}{h_i})\, dx = 1 \tag{3.5}$$

where Lbk, and Rbk denote the left and right boundary points of a span in which sample s_k falls, and S denotes the dimension of discrete space on node i. Given an

application-specific threshold, we consecutively select discrete bins, centered at s_k, over the span with rightmost and leftmost boundaries, Rb, Lb to satisfy

$$eq1 - \frac{1}{h_i}\int_{Lb}^{Rb} \phi_i(\frac{x - s_k}{h_i})\, dx \leq \sigma \tag{3.6}$$

To combine the kernel method with IPF, we can simply select

$$Rb - Lb = nB, \quad n < S \tag{3.7}$$

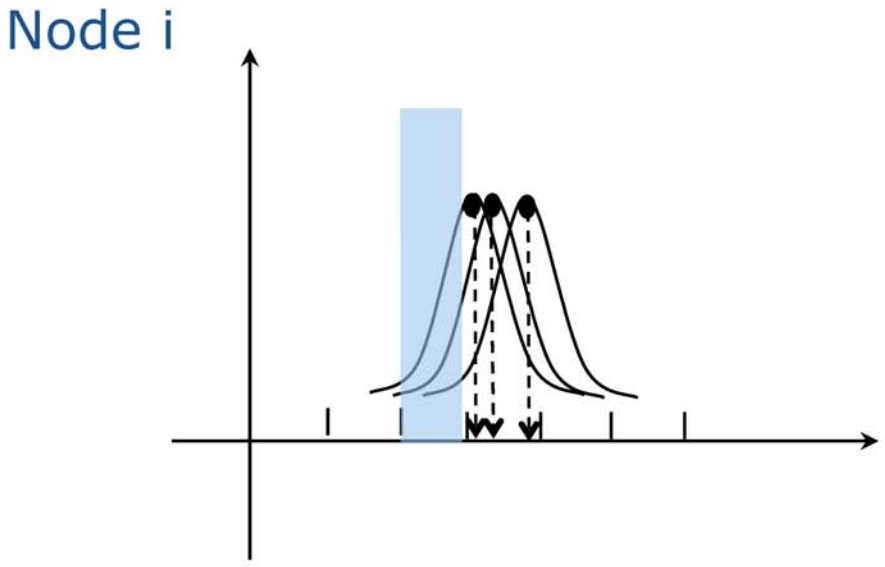


Figure 3.1. Illustration of kernel-based IPF.

where B is the span of one discrete bin, and n is the number of bins in the interval of [Lb, Rb]. Without loss of generality, we assume all the random variables have the same discrete space S. To learn the potentials attached to individual edges in a pairwise MRF model, we now need to express the correlation between two neighboring nodes with the weight assigned to each training sample by the smoothing kernel function. With i.i.d sampling of each random variable associated with a sensor node, to describe the kernel-based statistical correlation between two nodes, the probability of joint configuration over two neighboring nodes i, j now expands to n×n tabular values based on the appearance of each pair of training samples p, q on nodes i

and j respectively. To illustrate, let us set n=3 with a given σ, we can then get an expanded table for our kernel-based IPF, as shown in Table 3.1.1, in which all the nine possible product pairs are indicated in Figure 3.1.1 with solid arrow lines, each of which corresponding to a cell in Table 3.1.1. As a result, for each training sample s_k falling into [Lbk, Rbk], the contribution to the statistical joint configuration will be

$$\sum_{p-1\leq a\leq p+1, q-1\leq b\leq q+1} T_{a,b} \tag{3.8}$$

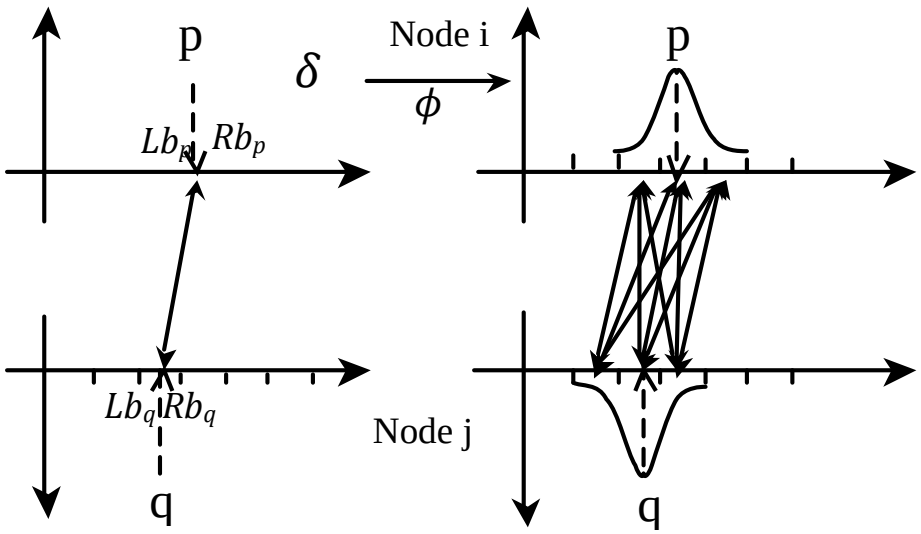


Figure 3.2. Expansion of kernel-based IPF.

In Figure 3.1.1, $T_(a, b)$ denotes the production in the expanded n×n table for cell (a, b), located at row a and column b, of the expanded table T for certain samples of any two adjacent nodes. A model with $n > 3$ or value of n equal to the number of states is a natural extension following the same rules. Such general case is implemented in the simulation of this paper where n is the same as the size of discrete space.

The key point of our kernel-based IPF is to appropriately smooth out each training sample over multiple discrete bins to maximally extract the statistical information one single sample can contain, so as to achieve the comparable training effectiveness

of the MRF model with less data samples. Thus it is important to tune the kernel function to correctly associate the weight with the covering range of a specific sample. That is, we need to choose an appropriate bandwidth h. Usually, the bandwidth can be computed following a target function, for example, the most common optimality criterion for bandwidth selection is the mean integrated squared error (MISE),

$$MISE(h) = E \int (\hat{f}_h - f)^2 \tag{3.9}$$

where $\hat{f}_h$ is the kernel estimator of f. Another popular choice is to measure the error with least square cross-validation, in the similar pattern. While there are various methods for automatic bandwidth selection, there is no theoretically satisfying approach for our specific application of kernel function for correlation learning, so we select a computational efficient method that can provide a stable performance based on our experiment results. Since we select Gaussian kernel function, to get the minimized MISE, we use the formula in [59]. In our approach, we first determine the range of bandwidth according to the selection of n, that is we can get

$$\frac{1}{2h} erf(\frac{x - s_i}{h})|_{LB}^{RB} - \frac{1}{2h} \geq 1 - \sigma \tag{3.10}$$

Then we choose the optimal h according to [59] as

$$h = (\frac{4}{d+2})^{\frac{1}{d+4}} N^{-\frac{1}{d+4}} \tag{3.11}$$

Table 3.1.
Expansion of kernel-based IPF in table

$\frac{1}{h_i}\phi_i(\frac{x-S_p}{h_i} = M_i^p P_i|_{Rb_p}^{Lb_p} = \int_{Rb_p}^{Lb_p} M_i^p)$

	$\int_{Rb_{p-1}}^{Lb_{p-1}}$	$\int_{Rb_p}^{Lb_p}$	$\int_{Rb_{p+1}}^{Lb_{p+1}}$
$\int_{Rb_{q-1}}^{Lb_{q-1}}$	$P_i\|_{Rb_{p-1}}^{Lb_{p-1}} P_j\|_{Rb_{q-1}}^{Lb_{q-1}}$	$P_i\|_{Rb_p}^{Lb_p} P_j\|_{Rb_{q-1}}^{Lb_{q-1}}$	$P_i\|_{Rb_{p+1}}^{Lb_{p+1}} P_j\|_{Rb_{q-1}}^{Lb_{q-1}}$
$\int_{Rb_q}^{Lb_q}$	$P_i\|_{Rb_{p-1}}^{Lb_{p-1}} P_j\|_{Rb_q}^{Lb_q}$	$P_i\|_{Rb_p}^{Lb_p} P_j\|_{Rb_q}^{Lb_q}$	$P_i\|_{Rb_{p+1}}^{Lb_{p+1}} P_j\|_{Rb_q}^{Lb_q}$
$\int_{Rb_{q+1}}^{Lb_{q+1}}$	$P_i\|_{Rb_{p-1}}^{Lb_{p-1}} P_j\|_{Rb_{q+1}}^{Lb_{q+1}}$	$P_i\|_{Rb_p}^{Lb_p} P_j\|_{Rb_{q+1}}^{Lb_{q+1}}$	$P_i\|_{Rb_{p+1}}^{Lb_{p+1}} P_j\|_{Rb_{q+1}}^{Lb_{q+1}}$

where N is the number of training samples and d denotes the dimension of kernel function. For example, when $N = 80$, we get $h = 0.4$. The bandwidth achieved with smallest sufficient sample size N usually provides optimal kernel function for IPF learning. With an appropriate kernel parameter (i.e., bandwidth) configuration, we will show that the proposed kernel-based IPF can provide significantly better learning results when the training samples are insufficient, in comparison to the traditional IPF learning.

3.1.2 Simulation and Analysis

For the convenience of understanding, the main flow of the kernel based parameter learning is illustrated in the similar pattern as basic probabilistic estimation in Figure 3.3. The major improvement is highlighted.

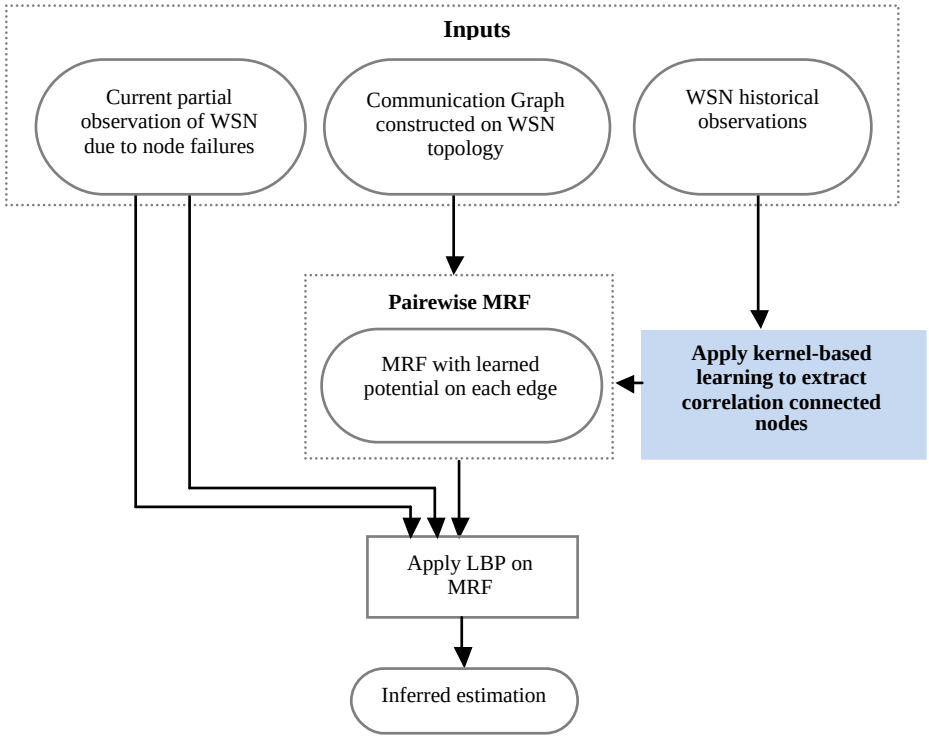


Figure 3.3. Flowchart of simulation of estimation with kernel based learning.

For the evaluation of Kernel-based Model, the same real-world sensing data from Intel Berkeley Research Lab is used. The sensor network of Intel Berkeley Research Lab consists of 54 Mica2Dot motes, operating on TinyOS, spreading over the whole lab, and we select 50 motes with enough temperature readings in this simulation. It is reasonable to assume that room temperature ranges from 15 to 30 degrees Celsius and thus can be descretized into 15 discrete states with the constant step of 1 degree. Based on the theoretical work of [68], the number of each motes neighbors necessary to maintain the connectivity of wireless networks should be in $\Theta(logM)$ where M is the total number of motes in the WSN. We choose the size of neighborhood as 5, close to when the constant in the formula equal to 1, and the resulting topology of the WSN (i.e., the WSN MRF model structure) is shown in Figure 3.4.

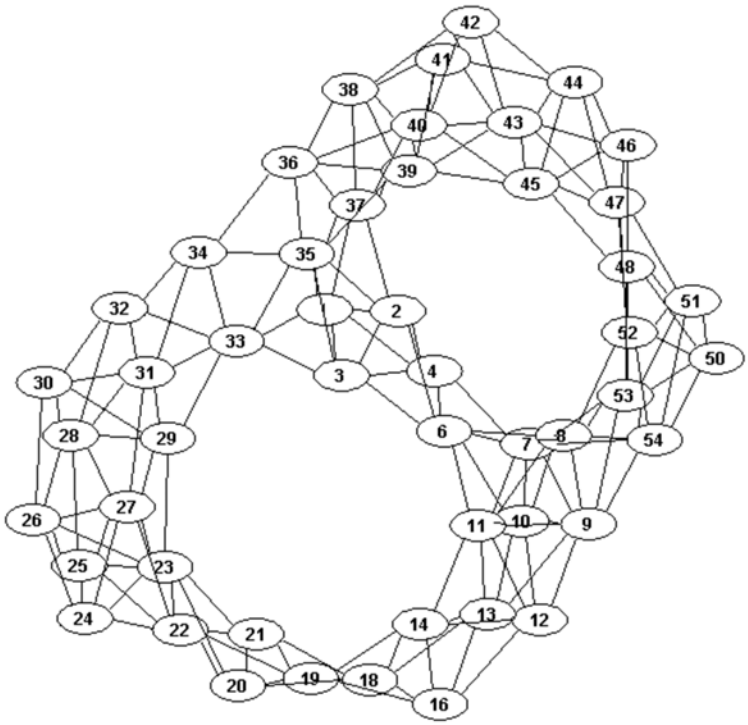


Figure 3.4. Network topology of the WSN for distributed statistical inference.

We use the same 80 sample sets for the MRF parameter learning process and 10 sample sets for validation (i.e. bandwidth selection) and testing respectively with both traditional IPF and our kernel-based IPF in the designated MRF model with selected neighborhood. For each test case, we randomly select a fraction of nodes as unobserved, and run LBP on the learned MRF model to get the estimation. To eliminate the influence of randomness to the evaluation of performance, we average the estimation over 30 runs for each test case, conducting 300 runs in total. One important parameter that can make great influence to the performance of a kernel model is the selection of bandwidth h. Based on the discussion of kernel methodology, the bandwidth can be decided as $h = 0.4$. However, it is just a theoretical selection and can not be applied to real-world application directly without further test. This bandwidth parameter selection is performed with validation process with 10 data samples. The performance comparison is illustrated in Figure 3.5 and Figure 3.6 in term of estimation accuracy rate and MAE. Based on these two figures, we can see that $h = 0.4$ is an valid choice and can get the optimal performance as expected.

The purpose of developing kernel based IPF is to make the parameter learning process, or the correlation information extraction, efficient in the term of training samples. Therefore we use partial training samples in the testing process with IPF and kernel based IPF respectively to see the comparison of the estimation performance: higher estimation accuracy directly suggests a more accurate model when all the other operations and parameters remain unchanged. Figure 3.7 shows the performance comparison for estimating missing data using our kernel-based IPF vs traditional IPF with 60 training data sets (For convenience, kernel-based IPF is written as KIPF). As we can see, KIPF offers much better performance as that of IPF, indicating that KIPF extracts more information from the limit training dataset. Now we evaluate our approach using different amounts of training data. As shown in Figure 3.8, when the number of training data sets is reduced, the estimation performance based on the MRF model learned by KIPF degrades, especially from 80 training samples to 60 of them. We note that when the size of training sets is reduced to 40 from 60, the

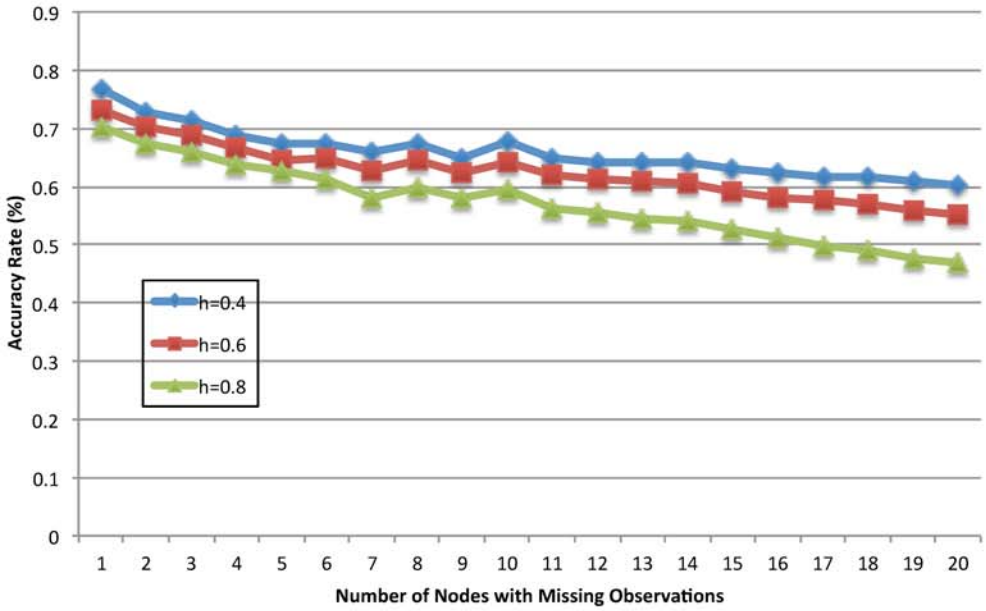


Figure 3.5. Comparison on the estimation performance with different bandwidth options (Accuracy Rate).

performance of KIPF is similar. When training dataset is insufficient, the performance comparison that really matters is between kernel and nonkernel models.

In Figure 3.7, the advantage of KIPF is clear over standard IPF in the estimation application. To further test the performance improvement of KIPF over IPF, various insufficient training sets are tested with KIPF and standard IPF. In Figure 3.9, the performance of KIPF shows huge advantage over IPF with the same number of training set (i.e. nTrain=40) and provide even a similar, if not better, performance as IPF with 80 training samples.

3.2 2D Kernel

3.2.1 2D Kernel Methodology

As has been illustrated in Section 3.1.1, we can handle limited training samples with 1D kernel during the parameter learning process. For that purpose, we need

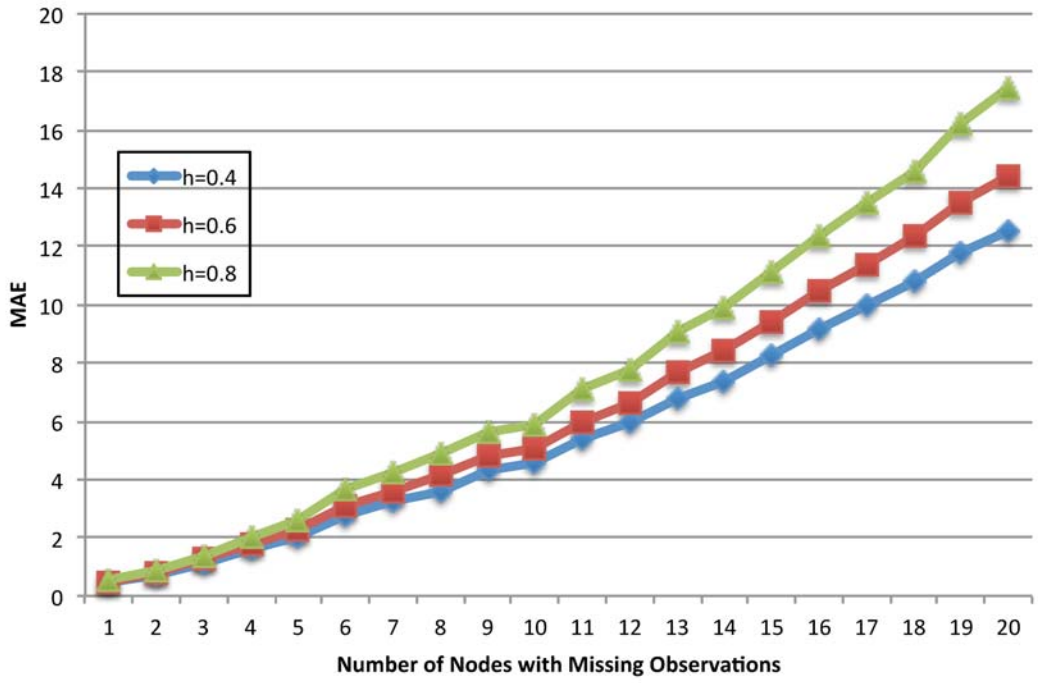


Figure 3.6. Comparison on the estimation performance with different bandwidth options (MAE).

to do kernel extension for each node on one edge, and establish a table to represent a 2D correlation. Although it is feasible, we need to decide multiple parameters in the process, which is a barrier for systematic solution, and the process is far from intuitive. Following the logic of kernel extension from the 1D solution, we want to move one step further to propose a more unified method, 2D kernel. When the 1D kernel computes the probability mass for each 1D kernel of two adjacent nodes, we project the problem to 2D space and use 2D kernel function to solve the correlation problem as whole. After the kernel density theories for continuous univariate data were introduced, they had been quickly extended to multivariate statistics. In 2000s, such multivariate kernel has been proved mature as its univariate counterpart. In our application, we will focus on 2D kernel. As we can consider multivariate kernel as

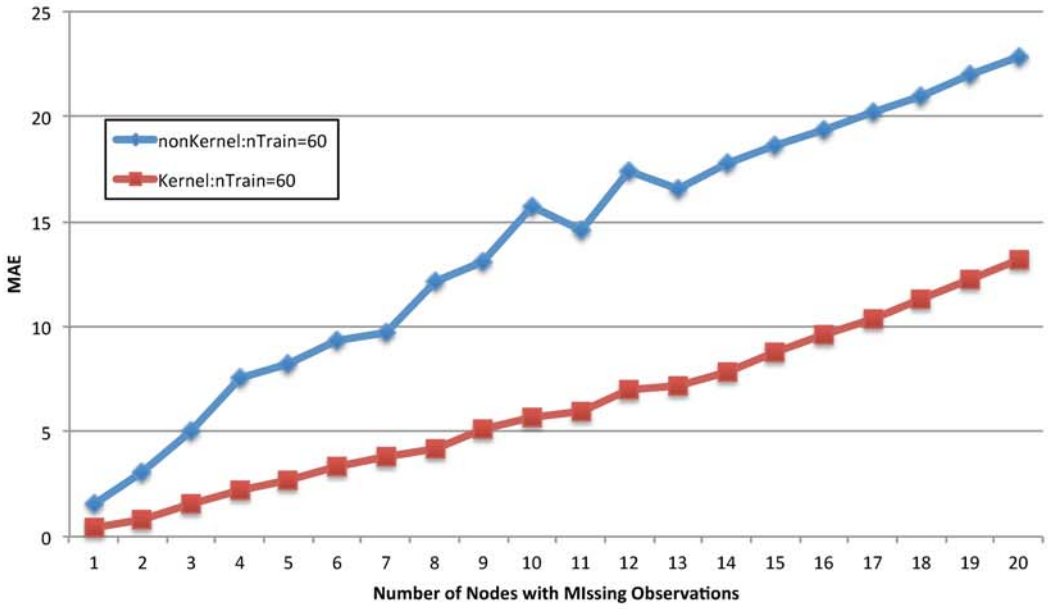


Figure 3.7. Comparison on the estimation performance between KIPF (h=0.4) and IPF with 60 training samples.

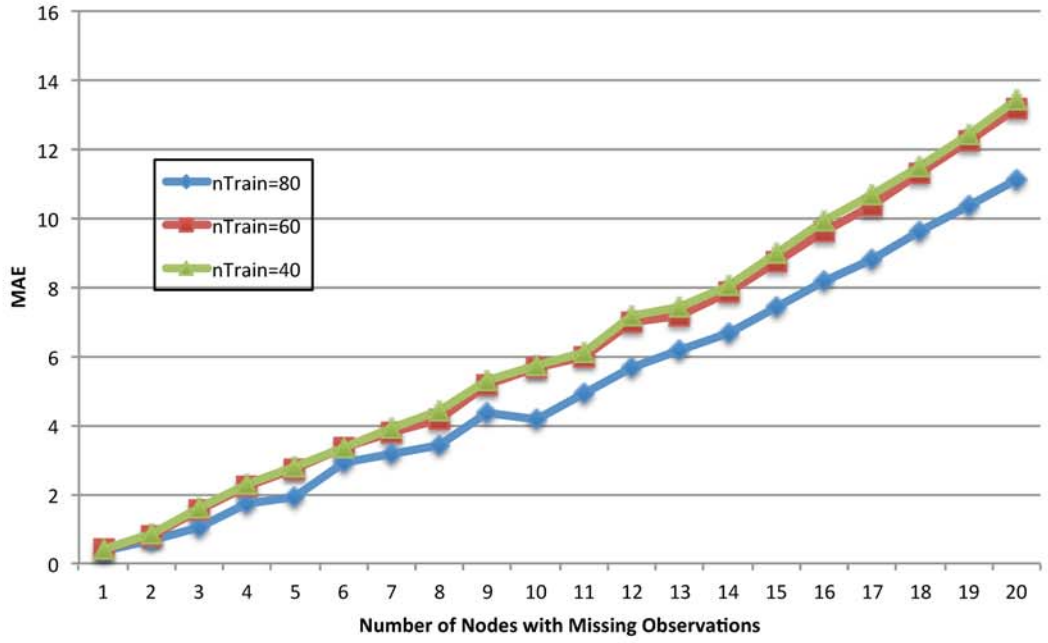


Figure 3.8. Performance of KIPF (h=0.4) with different number of training datasets.

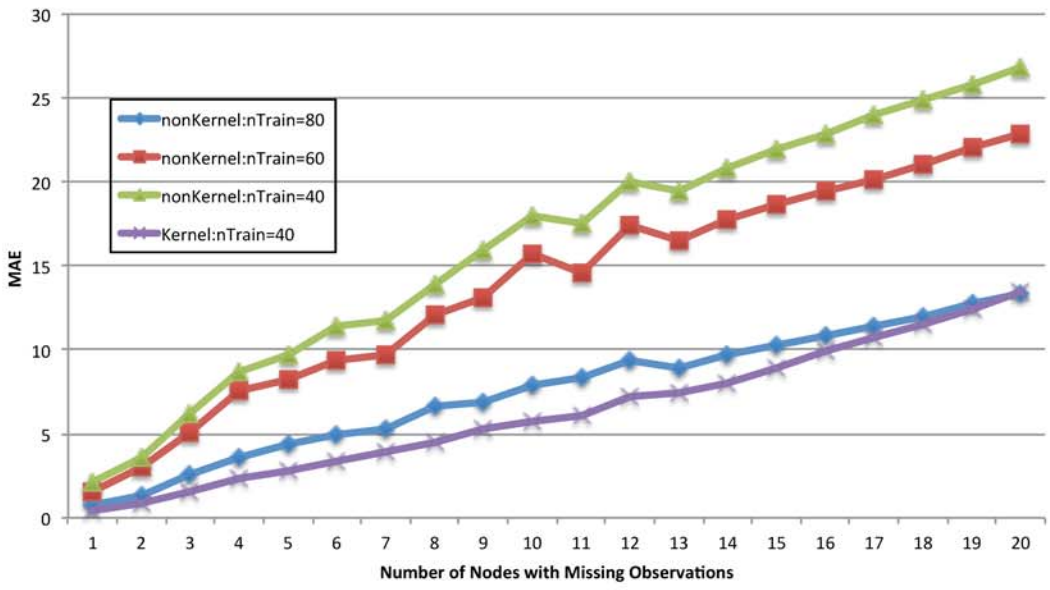


Figure 3.9. Performance comparison of KIPF (h=0.4, nTrain=40) with IPFs (nTrain=80, 60, 40).

a direct extension from 1D case, the basic idea is the same, and we can just use a multivariate kernel function, instead of 1D ones. That is:

$$\hat{p}_H = \frac{1}{n}\sum_{i=1}^{n} K_H(x - x_i) \tag{3.12}$$

$$K_H(x) = |H|^{-1/2} K(H^{-1/2}x) \tag{3.13}$$

where K denotes a multivariate kernel function and H is the vector of bandwidths. As we focus on 2D kernel, H will be the bandwidth matrix of size 2×2, which is symmetric and positive definite, corresponding to the vector of variables as $X = (x_1, x_2)^T$. Similar to the features of 1D kernel, the shape of 2D kernel function is not crucial to the estimation accuracy, but the choice of bandwidth dose. For convenience of processing, we use a popular choice, a 2D Gaussian function:

$$K(x) = \frac{1}{(2\pi)^{k/2}|\sum|^{1/2}} exp(-\frac{1}{2}(x-\mu)^T \sum^{-1}(x-\mu)) \tag{3.14}$$

As we have learned from 1D kernel functions, the most important factor than can affect the performance of kernel function is the selection of bandwidth. This

observation is still true for building 2D kernel function and we need to put most our effort on it. Actually, it is more crucial for 2D kernel function, comparing to 1D kernel, since its bandwidth matrix will decide not only the amount of smoothing, like 1D kernel bandwidth dose, but also decide a new feature: the orientation of the smoothing, which is the basic difference between 1D and 2D kernel functions.

Based on the smoothing orientation, there are three main categories for 2D kernel bandwidth matrix selection: 1) an identity matrix times a scalar; 2) a diagonal matrix with positive entries on the main diagonal; 3) a symmetric matrix with positive and definite entries. In a more intuitive way, the case 1 will result a 2D kernel density function with the same amount smoothing factor in all coordinate directions, which is the most simple one. We can basically consider that there is no direction in such case. In case 2, the complexity increases by allowing different smoothing factors in each coordinate. The case with the highest freedom and complexity is case 3, which allows arbitrary smoothing factor and orientation for 2D kernel functions. The first two cases, with less complexity, are actually efficient enough to handle most cases and are most widely applied though it has been proved that the further accuracy gain can be achieved by using case 3, the more general form.

In our simulation, we use case 2 bandwidth, which gives a better balance between complexity and accuracy. The basic criterion for different types of bandwidth matrix is the mean integrated squared error (MISE) between real density and estimated ones.

$$MISE(H) = E[\int (\hat{p}_H(x) - p(x))^2 \, \mathrm{d}x] \tag{3.15}$$

As this is not an expression with closed-form, an asymptotic approximation (AMISE) is usually necessary. We have discussed the difference between our application of kernel density estimation and the classic ones as we use it in a discrete space not a continuous one. That is the contribution for the sum of probability mass is calculated only by the discrete states: 1) the overall probability is computed by summing kernel mass from each data sample falling to a specific bin, i.e. a section [Lbk, Rbk] for 1D kernel function estimation; 2) the probability contributed by each data sample will only affect a predefine range, i.e. constrained to three adjacent bins for 1D kernel

function. As we included the uncertainty of each observation by 1D kernel function, we use the 2D density function to represent the ambiguity of correlation during parameter learning process. Following the designed 2D kernel function, we project an area on a 2D plane, when the shape of the area is decided by the features of kernel function and the boundaries of 2D bins. If we denote the two coordinates as X and Y, then each square in a grid can be denoted as Area: $[LB_X, RB_X], [LB_Y, RB_Y]$. During the kernel estimation, each sample combination x, y will contribute amount of probability mass for a predefined area. In our simulation, we project the 2D kernel over the whole available area (minX, maxX), (minY, maxY). For the 2D normal kernel function and category of bandwidth matrix we choose, there are two smoothing factors along X and Y coordinates, decided solely by the optimized bandwidth matrix learned from training samples, as shown in the Figure 3.10.

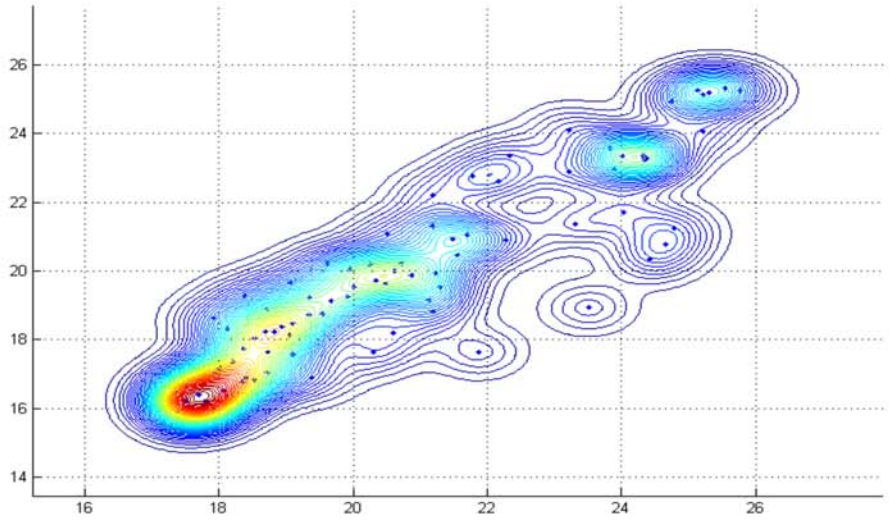


Figure 3.10. 2D kernel density estimation on data distribution.

One thing worth noticing is that the bandwidth matrix on each edge of the MRF model is different since we have different set of training data for each edge. Regardless the values of two smoothing factors, each data sample will have more effect to its

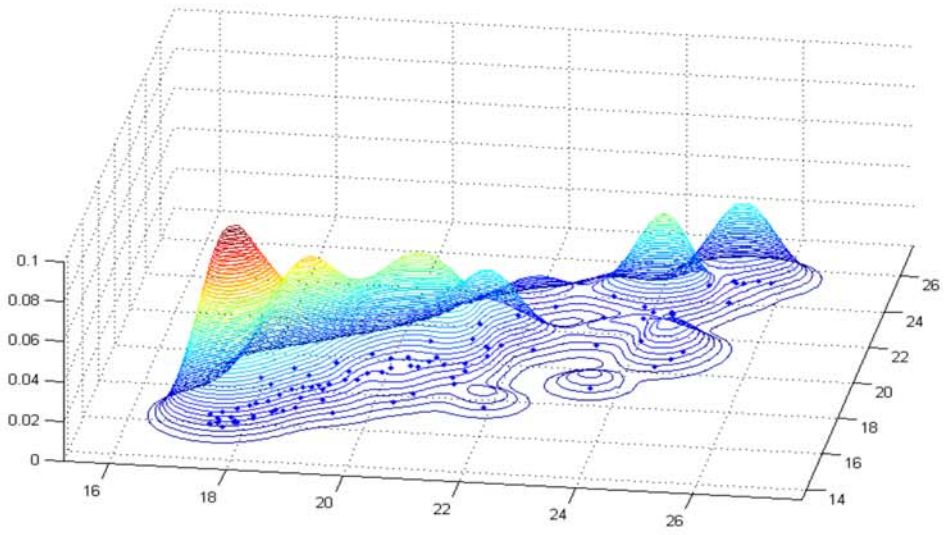


Figure 3.11. 2D kernel density dstimation on probability distribution.

adjacent squares and the influence will reduce rapidly when the distance to other squares goes further. The degradation speed of influence caused by each sample data will be decided by the smoothing factor, or more specifically, the bandwidth matrix. Comparing to the 1D kernel applied in the previous chapter, we actually make no assumption that the uncertainty of two adjacent nodes are I.I.D, like in the 1D case, so it can be used concern-free. Further more, when we solve the problem in 2D space, it naturally fits for summing up 2D uncertainty and it is unnecessary to convert the 1D probability to 2D following a 2D table.

3.2.2 2D Kernel Simulation-Data

For the purpose of comparison, we perform both 1D and 2D kernel to the same WSN data following the similar workflow as introduced in Section 3.1. For 2D kernel, the only difference is the process of building kernel function. We use experience formula to decide the value of bandwidth in 1D kernel function, in which all the bandwidths are the same for difference edges. In 2D kernel case, the bandwidth

matrixes are decided by training samples from each edge, so they are different by edge. A real-world data set is used in our empirical study. The first data set comes from the Southern Great Plains Hydrology experiment of 1997 (SGP97) in Oklahoma, in which volumetric soil moisture content for the top 5 cm of soil were derived at 800-meter resolution by inversion of L-band microwave radiobrightness temperature images retrieved with electronically scanned thinned array radiometer (ESTAR) ([38], [48]). In this data set, there were 17 data images of such ESTAR data corresponding to 17 different days within the period ranging from June 18 through July 16 of 1997. The 17 spatial structures of the near-surface volumetric soil moisture content, in a unit of m3/m3, were retrieved from the ESTAR data. In each image, red area represents wetter soil moisture while blue area represents drier soil moisture in the top 5 cm surface soil layer. As shown in the moisture distribution graph, the 14 sets of the volumetric soil moisture content (represented in percent) retrieved from ESTAR data are used as the training data in our experiment, while other three sets of the retrieved volumetric soil moisture content data are used as test data to validate our approach. One of the important features in this data set is that there was no rainfall occurrences on any of the 14 days used in the training data sets, while a relatively large rainfall event occurred, one day before the July 16 test data. These rainfall events play very important roles in the re-distribution of the spatial structure of the top 5 cm volumetric soil moisture content.

3.2.3 2D Kernel Simulation-Modeling

Considering the physical configuration of remote sensing, each point intuitively represents an environmental variable (i.e., soil moisture or vegetation) in a pairwise MRF model. A pairwise potential function coding the correlation relationship between any two adjacent sensing points in either latitude or longitude direction is associated to an edge clique, which is encoded as the first-order markovian property of the modeled pairwise MRF. Conditioned on its directly connected neighbors, a

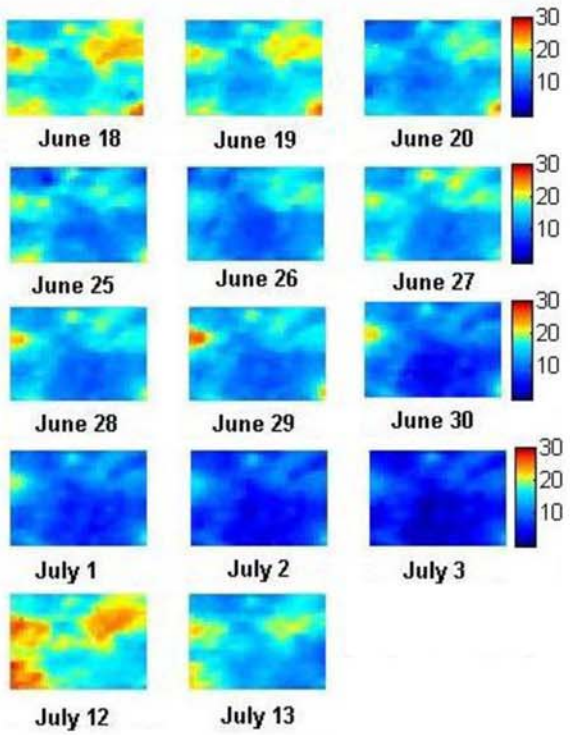


Figure 3.12. Spatial distributions of training data.

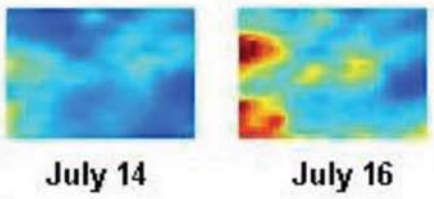


Figure 3.13. Spatial distributions of two test data sets.

sensing point is independent of other remote sensing points. In this empirical study, the dimension of our pairwise MRF model is 32×32, modeling the grids of 32×32 remote sensing variables for both real-world data sets. As showed in Figure 3.14, each arrow represents a correlation function to be learned from the training data set and all correlation functions actually have directions in terms of realization of the MRF learning. This configuration allows us to obtain the correct local beliefs and belief messages during the LBP, and at the same time, to cut down the unnecessary com-

putation by avoiding repeating multiplication for the same potential function twice for different directions along an edge to which the potential is attached. That is, according to the propagation rules of the LBP, each edge needs to transmit belief messages at least twice in different directions, we can just compute the belief propagation (belief messages multiply the corresponding potential function) in one specified direction, and achieve the one in the opposite direction accordingly.

The values of the given available real-world soil moisture training data fall into the range of [0, 28], so we discretized the space evenly to 14 states (denoted as a14-state model). Note that ECH2O has a typical accuracy of (volumetric soil moisture content in percent) on all soils, and (volumetric soil moisture content in percent) with soil specific calibration. Therefore, the discretization error produced by the 14-state model (i.e., about) in our experiment is consistent with the margin of error specified on the typical sensors data sheet. That is, our experiment on the 14-state model has practical significance and the constructed MRF model can be directly applied to real-world tasks. Therefore, we will focus more on the 14-state MRF modeling in our empirical study. On the other hand, one can imagine that constructing any 14-state MRF model with only 14 training data samples would be quite difficult. Thus, our empirical study of kernel based learning with the soil moisture data set represents a challenging test on the proposed approach with minimal real-world environmental monitoring data as the training data in MRF learning.

3.2.4 Simulation and Analysis

In our experiment, unobserved points are randomly distributed in each trial. For a given percentage of unobserved points, to make our empirical results statistically reliable, 20 trials are conducted and the average performance of the 20 trials is calculated. For only 14 training samples, it can be safely considered a case with insufficient training samples and implementation of kernel method is a natural choice. Based on the discussion of 1D and 2D kernels, the bandwidth of 0.6 is selected for 1D model

and case 2 bandwidth is applied for 2D kernel. The results for the 14-state model are shown in Figure 3.15 and Figure 3.16 respectively for both accuracy and MAE.

Figures 3.15 and 3.16 show the evaluation results using test data of July 14 (dry day). The vertical axis presents estimation accuracy rate measured by the rate of the correctly estimated data points out of the total unobserved points. From Figures 3.15 and 3.16, we can see that (1) kernel based methods generally produce better performance than that of standard IPF learning, and (2) 2D kernel generally produce better performance than 1D kernel method in terms of both accuracy rate and MAE. Improvement on the performance of estimation on 2D kernel model over 1D kernel and standard IPF learned network demonstrated clearly shows that the some important spatial correlation relationships missed in 1D kernel method are captured successfully by the 2D kernel method and then fused by the belief inference in the MRF models. To draw a more concrete conclusion, the same performance comparison is also conducted with another representative case, July 16 (wet day) as below. The result

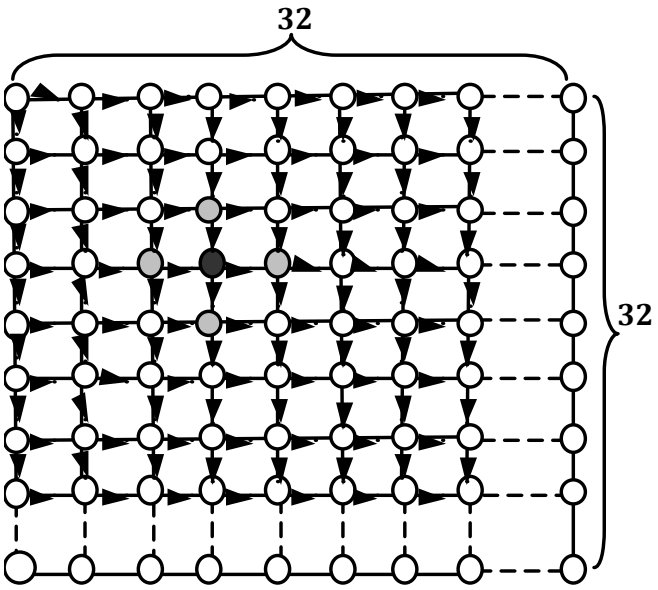


Figure 3.14. LBP realization on pairwise MRF model for sensed data grid.

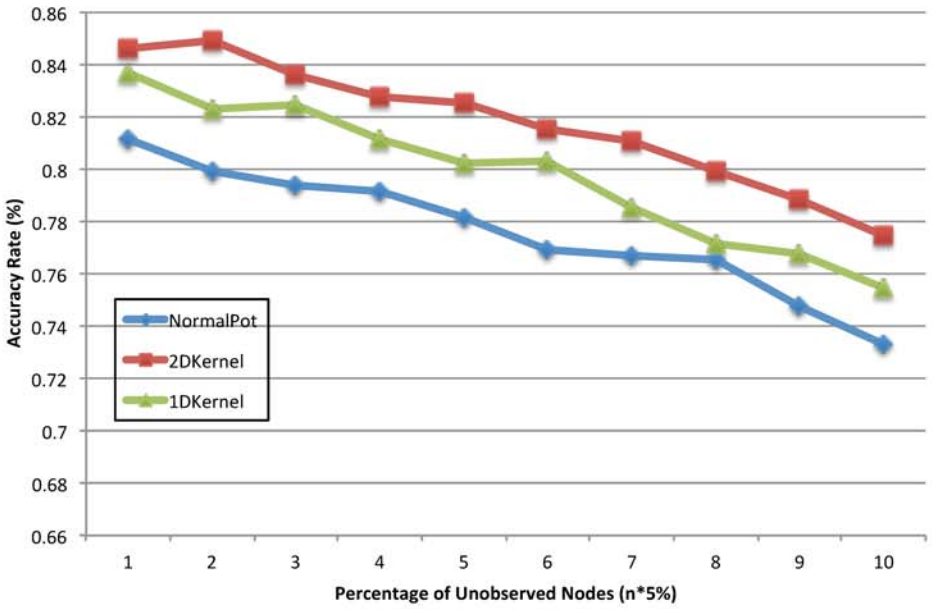


Figure 3.15. Estimation performance in terms of accuracy rate with the standard model, 1D and 2D kernel models (Dry, 20 trials).

analysis are illustrated in Figure 3.17 and Figure 3.18 respectively in term of accuracy rate and MAE.

As expected, the overall performance achieved by all models for wet day are worse than the dry day considering that more information is necessary for correct estimation of more uneven distribution of soil moisture in wet day. However, the relative improvement on the performance of kernel based learning over standard IPF is actually quite evident in both terms of accuracy rate and MAE. More importantly, 2D kernel shows a clear advantage over 1D kernel for missing observation estimation task in this case. Furthermore, 2D kernel method provides more robust performance when the percentage of missing observations increases. For both wet and dry data, 2D kernel produces a smoother curve comparing to 1D kernel.

Since bandwidth is an essential parameter that has direct impact to the estimation performance, more experiments have been done to exam the robustness of the performance in relation to the selection of different bandwidth matrixes. Based on the

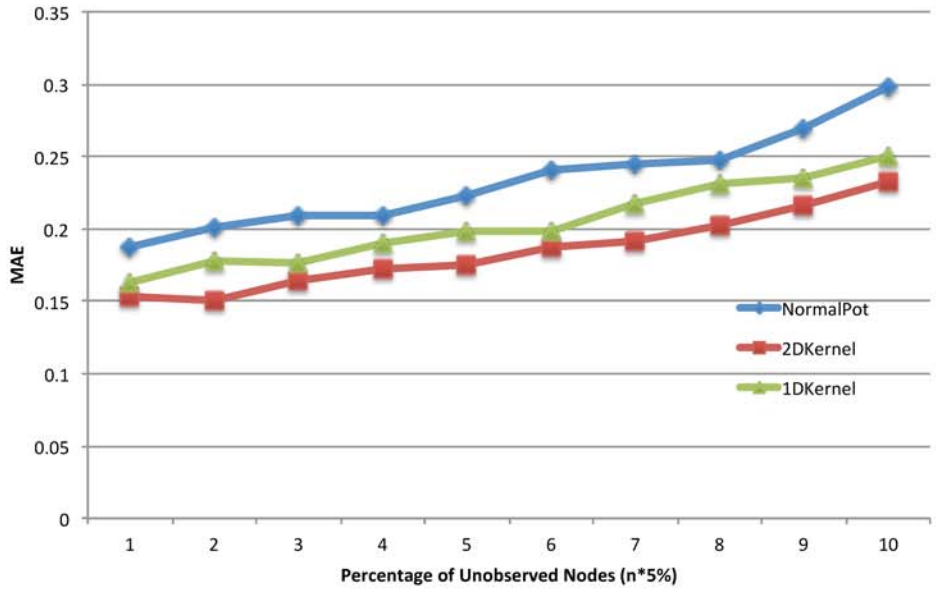


Figure 3.16. Estimation performance in terms of MAE with the standard model, 1D and 2D kernel models (Dry, 20 trials).

computed optimal bandwidth, we keep adding scalar value to the primary diagonal and exam the estimation performance of inference to evaluate the robustness of 2D kernel model. The scalar value will start from relatively small value comparing to the selected matrix.

In our simulation, different edges need different bandwidth matrixes to best fit the training data collected from both of two ends and the range for entries along main diagonal is [1, 2] so we start the scalar value from 0.1. As shown in Figure 3.19 and Figure 3.20 for dry test case, the performance has good robustness to different bandwidths when the scalar changes by 0.1. The performance curve indicated as 2D Kernel is based on 2D kernel model built with optimal bandwidth computed. For both accuracy rate and MAE criterions, the estimation performance with low percentage missing rates will vary slight with different bandwidths but the overall performance is almost the same. It shows that even though the bandwidth is important, the concern to get exactly correct bandwidth to ensure the expected performance is not necessary

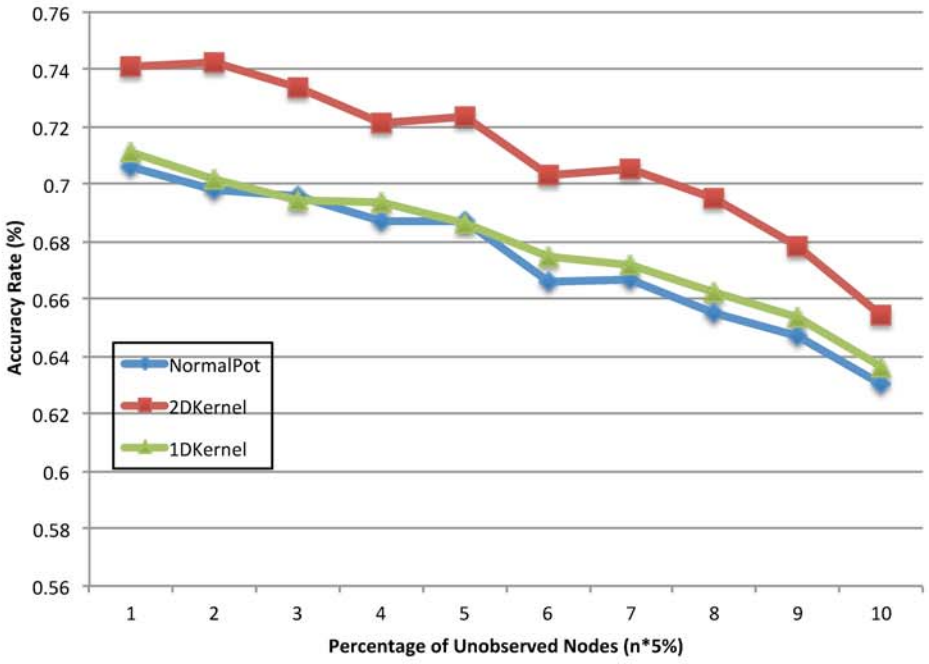


Figure 3.17. Estimation performance in terms of accuracy rate with the standard model, 1D and 2D kernel models (Wet, 20 trials).

and there is sufficient robustness margin in relation to selection of bandwidth for 2D kernel model.

For further exam on the robustness, we increase the scalar to 0.5 for the same test case and test the performance when bandwidth changes in both directions: increase and decrease. The results are illustrated in Figure 3.21 and Figure 3.22. In term of either accuracy and MAE, the 2D kernel model shows robustness to change of bandwidth along primary diagonal. In comparison to Figure 3.19 and Figure 3.20, these two figures show a little more variation but the tendency is the same: the performance almost keeps the same with continuous changes to bandwidth. For wet test case, we can get the same conclusion as shown in Figure 3.23 and Figure 3.24.

The illustrated robustness to bandwidth selection will not diminish the importance of parameter λ because we always need to find the appropriate range of bandwidth and the robustness of 2D model just facilitate such selection process: it will be easier

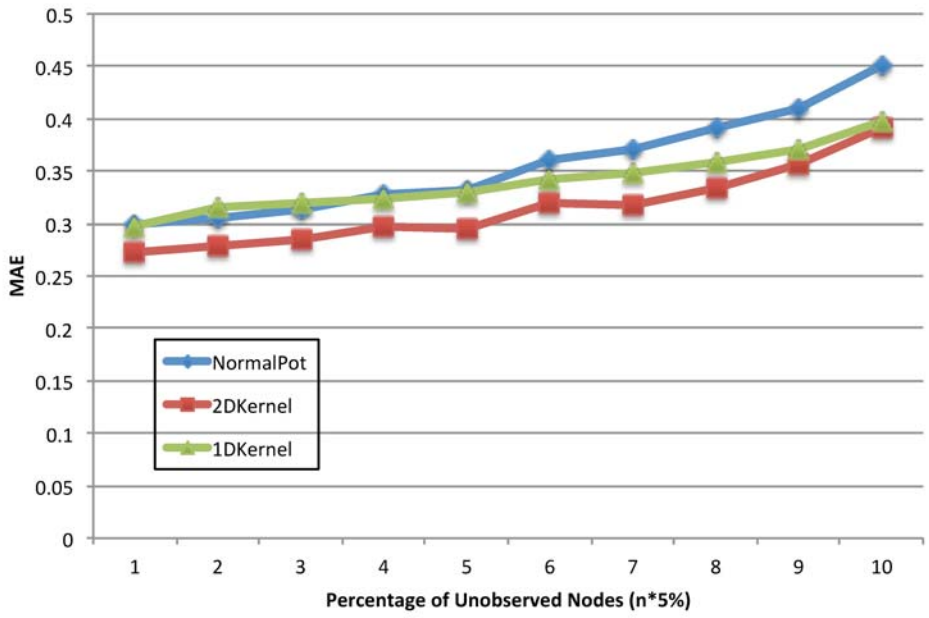


Figure 3.18. Estimation performance in terms of MAE with the standard model, 1D and 2D kernel models (Wet, 20 trials).

to find a range than specific value. If the bandwidth falls in inappropriate range, the negative impact will be immediately reflected by the degradation of the estimation performance as shown in Figure 3.25 and Figure 3.26. We can see that the performance of 2D kernel drops below the model with normal potential when the implemented bandwidth diverge from the optimal bandwidth.

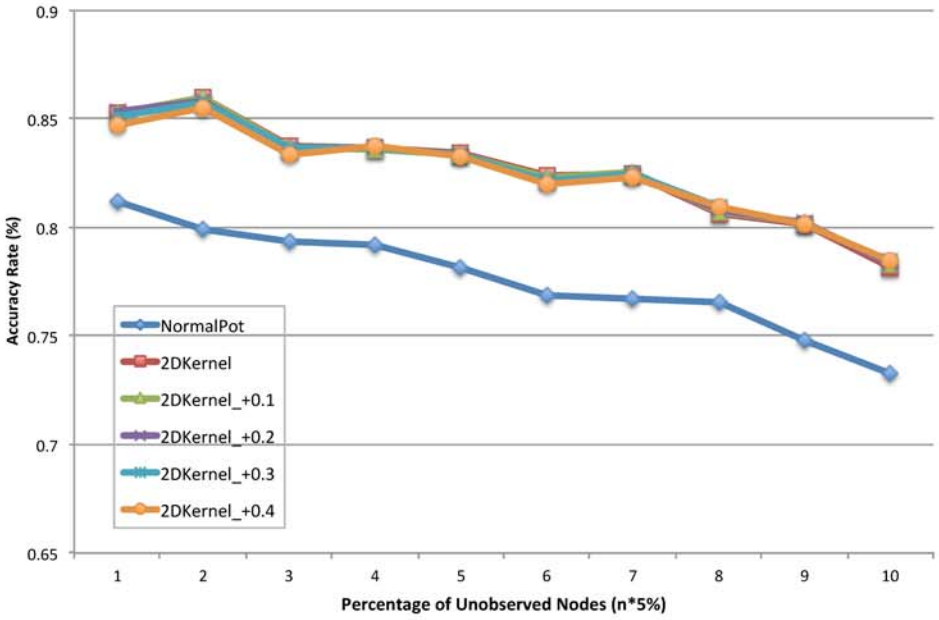


Figure 3.19. Robustness analysis based on estimation performance with 2D kernel models (Dry, Accuracy Rate, Increase by 0.1).

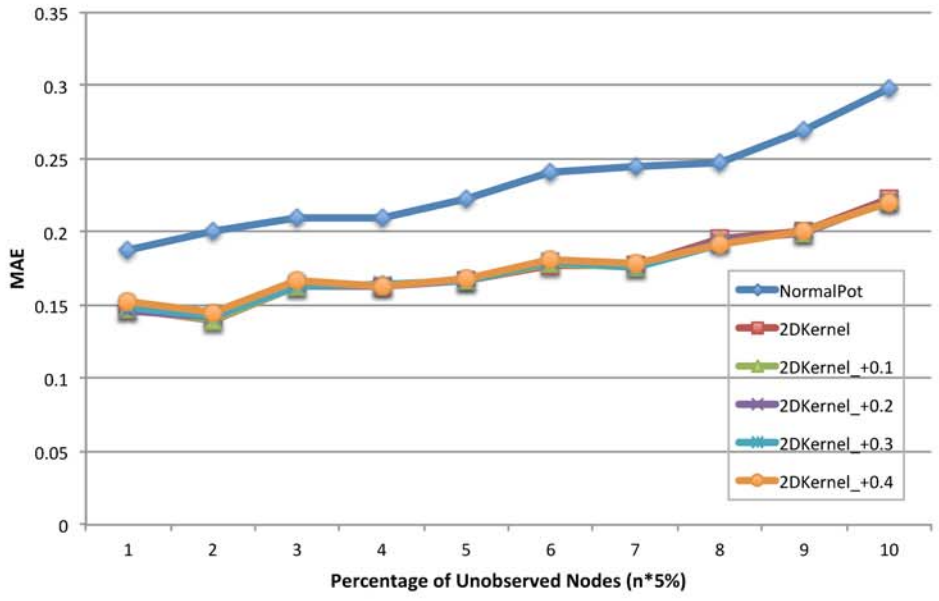


Figure 3.20. Robustness Analysis based on estimation performance with 2D kernel models (Dry, MAE, Increase by 0.1).

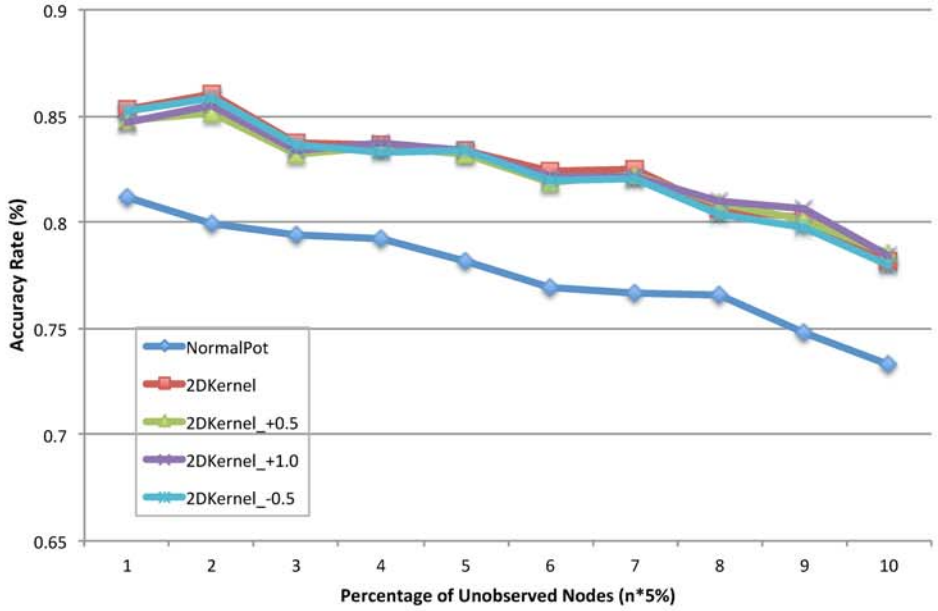


Figure 3.21. Robustness analysis based on estimation performance with 2D kernel models (Dry, Accuracy Rate, Change by 0.5).

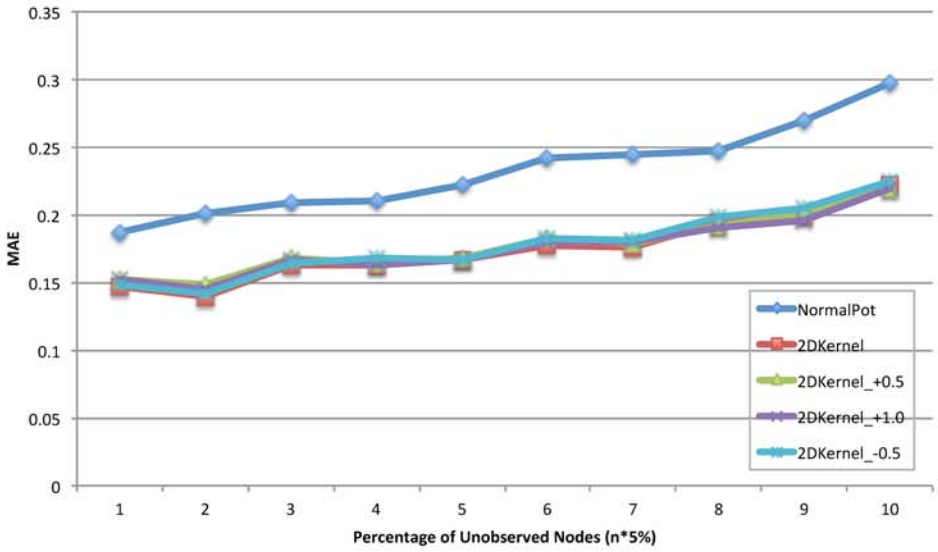


Figure 3.22. Robustness analysis based on estimation performance with 2D kernel models (Dry, MAE, Change by 0.5).

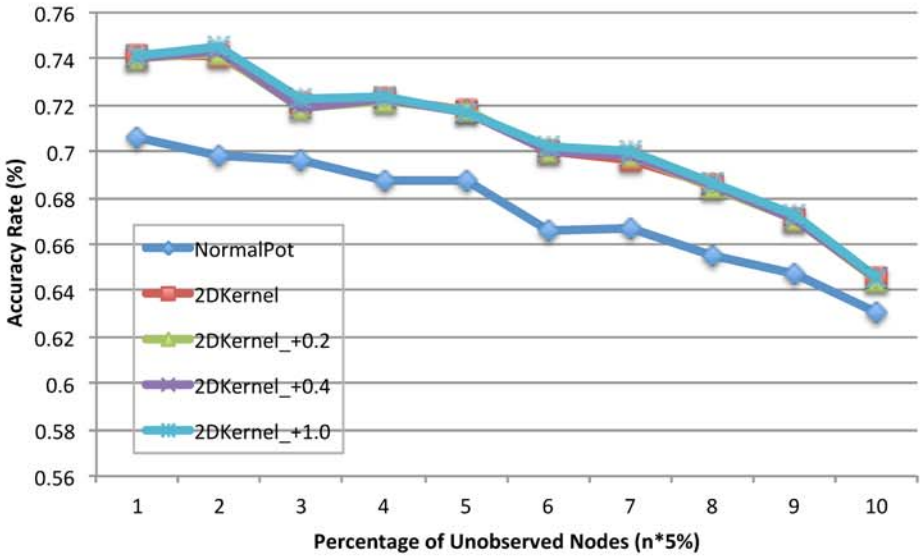


Figure 3.23. Robustness analysis based on estimation performance with 2D kernel models (Wet, Accuracy Rate).

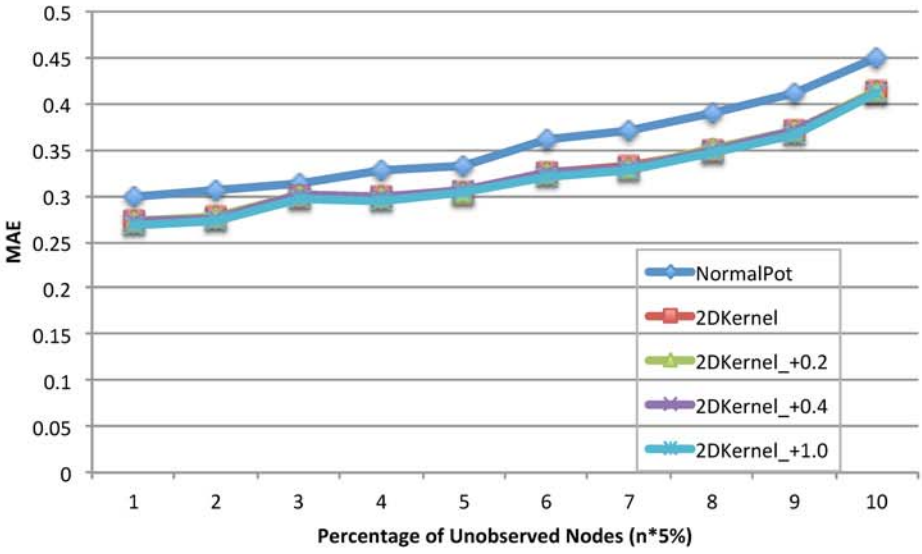


Figure 3.24. Robustness analysis based on estimation performance with 2D kernel models (Wet, MAE).

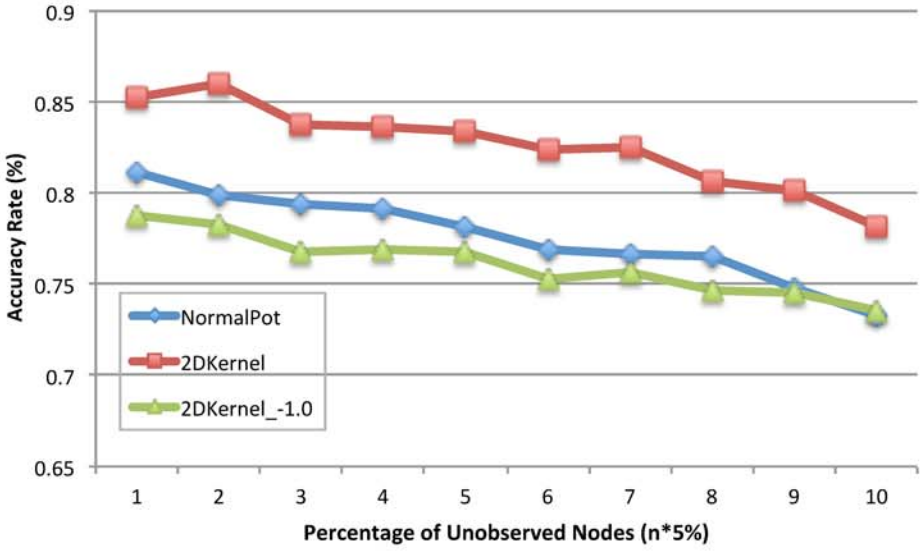


Figure 3.25. Robustness analysis based on estimation performance with 2D kernel models (Dry, Accuracy Rate, Inappropriate range).

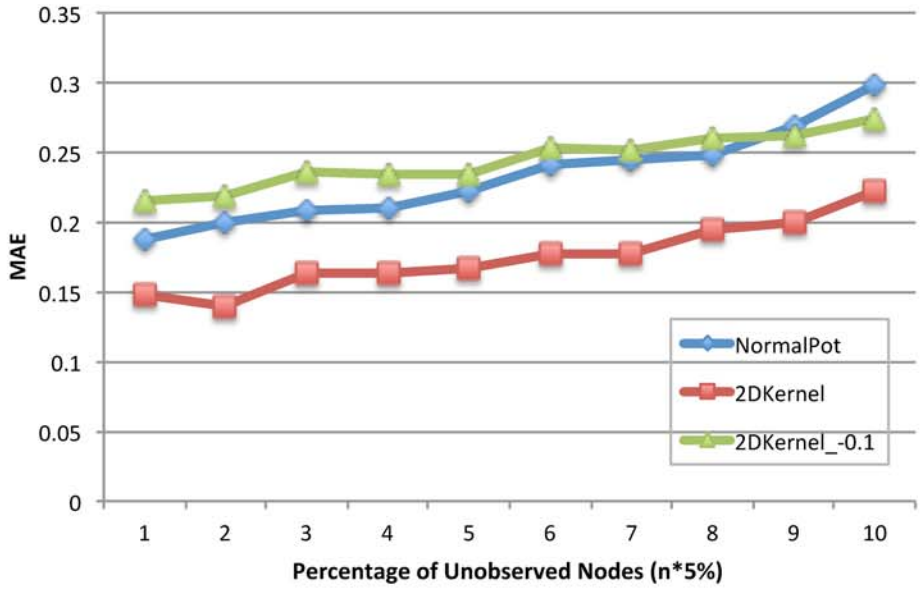


Figure 3.26. Robustness analysis based on estimation performance with 2D kernel models (Dry, MAE, Inappropriate range).

4 INFORMATION GRAPH

4.1 Methodology

As the data communication contributes the most to the energy consumption, we are motivated to perform the estimation/belief inference on a most possibly simplified network structure, instead of using the communication network directly([76]). The less edges, and more importantly the less loops, exist in the tailored network, the less belief messages transmission is required and the faster the belief propagation will converge, both of which will greatly benefit the energy efficiency of a WSN.

Our approach starts with a robust WSN CG and constructs an MRF DG as a sparse subgraph of the given CG through data-driven graphical model optimization. Connectivity is a critical question for WSNs, which can be usually modeled as, without loss of generality, a random k-nearest neighbor graph (or random geometric graph) G(V, E(k)) ([50], [68], [3]), where V is the vertex set, k is the number of the nearest neighbor nodes with two-way connections to each vertex, and E(k) is the corresponding edge set of G. In this paper, we adopt the notion of random k-nearest neighbor graph G(V, E(k)) to model a robust WSN connectivity graph with an appropriate k. As a subgraph of the CG, DG makes the no-routing property naturally hold, where message-passing on any edge of DG for in-network inference will be a one-hop communication in the WSNs underlying CG. As an concrete example, the DG and CG relationship shown in Figure 4.1 can give a better picture of DG reduction. On the other hand, since the constructed information model DG is a sparse subgraph of the CG, the number of messages needed in the WSN message-passing inference based on the DG can be significantly reduced in comparison to those directly based on the original CG of the WSN. Consequently, the energy consumption of distributed in-network inference in the WSN can be significantly reduced.

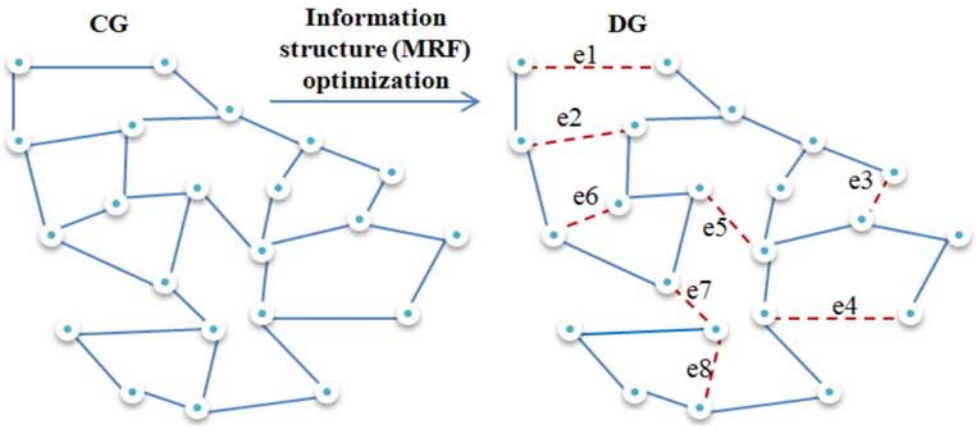


Figure 4.1. Inference in DG vs inference in CG (with network topologies).

To begin, the first question to address is how to construct an efficient and robust CG of a WSN, that is, how to determine the appropriate number of k in connectivity graph G(V, E(k)). The robustness of CG should ensure the connectivity of the WSN in dynamic communication environments. Based on the theoretical work of [68], the number of neighbors of each mote necessary to maintain the robust connectivity of wireless networks as the size of WSN increases should be in $\Theta(logn)$ where n is the total number of sensor nodes/motes in the WSN. The work of [68] shows that the neighbors for each mote should be in the range from 0.074log n to 5.1774log n. Later, [3] improves these lower and upper bounds to 0.3043log n and 0.5139log n respectively. We use $k = logn$ to determine the appropriate neighborhood size k in the CG G(V, E(k)) to maintain a robust WSN connectivity. Since packet transmission and reception contribute the major part of WSN energy consumption, one can typically select the adjacent neighbors with the shortest transmission distances to build CG, consistent with the k-nearest neighbor graph model, which is reasonable if no prior knowledge about individual channel qualities is available.

Once the CG of G(V, E(k)) is obtained with $k = logn$ given the WSN size of n, the major question is how to construct an optimal DG from the obtained CG. From the information modeling perspective, the goal is to remove those edges in the CG

with weak and noisy correlations between the connected nodes. Consequently, the distributed in-network inference within the WSN performed on the constructed DG, instead of directly on the WSN CG, leads to the reduction of energy consumptions. Thus, our proposed information modeling approach can be formulated as: given a CG, maximize DG model correlation fitness subject to the following two constraints: (1) DG being sparse; and (2) DG being connected. This way, a sparse (and robust) connected DG can be constructed from the CG, whose overall information correlation fitness is maximized among all possible information model candidates with the same sparseness. The graphical model topology learning is based on recent studies of graphical model optimization in machine learning. Our proposed approach is outlined in Figure 4.2. Currently our information modeling approach is an offline learning process due to its computation cost.

Our research shows that in-network inference on the constructed sparse DG provides advantages over the original CG in the following aspects: 1) dramatically reduces the complexity of message-passing to save energy; and 2) contains fewer short cycles and thus improves the robustness of BP-based inference approaches. In addition, our DG structure optimization approach is orthogonal to existing methods of improving BP for inference in WSNs, and thus can be applied to WSNs in combination with those existing methods.

We use the formula $NeighborSize = logN$ to determine the appropriate size of a neighborhood in building the CG of a WSN. Since packet transmission contributes the major part of WSN energy consumption, we select the adjacent neighbors with the shortest transmission distances to build our CG, which is consistent with the geometric random graph model. The major question is how to construct an optimal DG from the obtained CG in our first step described above. Basically, the goal is to remove those edges in the CG with weaker correlations between the connected nodes, through DG structure optimization. This way, the obtained $DG(V, \acute{E})$ is a subgraph of $CG(V, E)$, where $\acute{E} \subset E$. We point out that the BP inference on the reduced DG provides advantages over that of the original CG in both the aspects of energy-

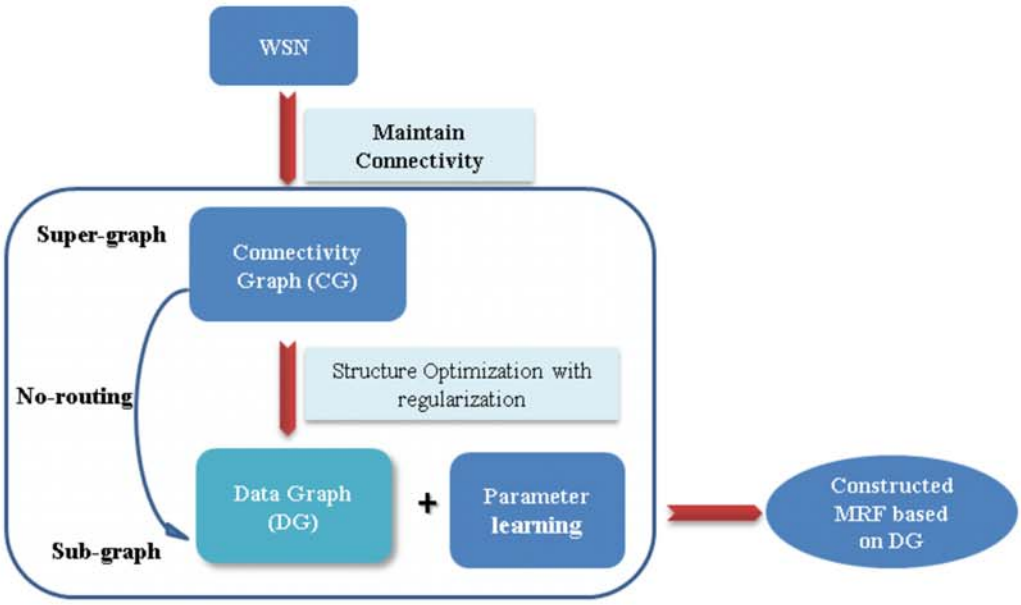


Figure 4.2. An outline of our approach to information structure optimization for distributed in-network inference in WSNs.

efficiency and performance robustness. To achieve an effective DG, our approach includes two aspects: graphical model topology learning and its parameter learning. The graphical model topology learning is based on recent studies of graphical model optimization in machine learning area.

When a graphical model framework is used for inference task, it is necessary to specify the structure of the model and the potential functions associated with each clique.

For a dense WSN for monitoring and surveillance, a correlation relationship exists among nodes in adjacent locations [29], and, such dependence information can greatly benefit the estimation performance if we can extract it as prior knowledge for the inference process. This requirement suggests that a data-driven structure be usually preferable to a graphical model application in WSNs.

Suppose given a collection $D = x^{(1)}, x^{(2)}, \ldots, x^{(k)}$ of k samples of a set of p random variables $X = x_{i=1:p}$. Let w denote a vector of parameters of pairwise MRF G(V, E) over the p random variables. Basically, w is a vector, whose elements, if nonzero,

are indexed by pairs of vertices (i.e., edges) of the underlying graph G. The goal of graphical model structure learning is to optimally infer the edge set E from the given collection of k samples. To accomplish this, two existing major approaches used are score-based approach and regression-based approach. The score-based approach is to find the model structure that can best fit the independence of nodes according to a scoring function measuring the fitness of the graph to training samples. However, for an MRF model, the scoring function will involve the computation of normalization constant, and will become computationally intractable, so its application is restricted merely to those simple networks with special structure properties, such as polytrees and bounded tree-width hypertrees.

In this work, we adopt the regression-based approach that has been gaining more recent attention [52], in which MRF model structure learning is formulated as a parameterized optimization problem with guaranteed global optimum and much better scalability. In the regression-based approach, the task of regression is to find parameters/weights w such that the best recovering of the edge set E can be achieved. In general, to avoid over-fitting, regularization is commonly employed, by which a penalty function (e.g., an L1 penalty) is imposed on the parameters w of the model under consideration. Furthermore, L1 (Lasso) regularization can lead to sparse graphical structure outcome. The sparseness of the learned model structure is desirable as it significantly simplifies the resulting MRF structure. Such regularized structure learning procedure has been proven successful in both Gaussian ([52], [45]) and discrete MRFs ([55], [52], [56]) The process of DG structure learning is illustrated in principle with pseudo code.

To explain lines 5-6 in the illustrated algorithm, let us consider the optimization with regularization in a pairwise MRF structure learning for constructing DG. Since our goal is to learn the network structure of an MRF, only the potential functions on edges will be involved and the node potential Φ_i can be set to 1. Then the MRF model can be presented in a log-linear model ([28], [27]):

Notations:

CG: the topology of CG, including all the nodes and edges

DG: the topology of DG, including all the nodes and edges

TrainD: training data set

threshold: the lower bound of correlation attached to an edge

weight(e): the strength of correlation attached to edge e

$\boldsymbol{\lambda_0}$**:** the initial value of $\boldsymbol{\lambda}$

stepsize: step length of $\boldsymbol{\lambda}$

Init(): Initiation procedure

Edge(G): the set of edges of G

Data_Graph_Structure_Learning (CG, λ_0, $threshold$, $TrainD$)

```
1: Init();
2: λ ← λ0;
3: while (1) do
4:    DG ← CG;
5:    for (Edge(CG)) do Search weights fit to all edges with the
6:        regularization; end for
7:    for (Edge(CG)) do
8:        if (weight(e) < threshold ∧ DG-{e} is connected)
9:            Remove the corresponding edge e from the DG; end for
10:   if (DG is sparse) break
11:   else λ = λ +stepsize;
12: end while
13: return(DG)
```

$$P(x, w) = exp(\sum_{c \in C} w_c f_c - A(x, w_c)) \tag{4.1}$$

where each $w_c(nonzero)$ is associated with a feature function f_c on a pairwise clique c (i.e., an edge). This function illustrated that the global probability over the whole network can be determined by the combination of weights associated with each configuration over one clique/edge. Each feature function is associated with one clique/edge c and represents all possible configurations of it. Thus, we can represent $logP(x, w)$ with a log linear combination similar to a linear model. As a special case of log-

regression, the detailed introduction of log-linear model can be found in [6]. To form the objective function, we get negative log-likelihood function as:

$$NLL(w) = \sum_{c \in C} -w_c f_c + A(x, w_c) \tag{4.2}$$

Then, the unconstrained objective optimization function can be formulated in a general form as

$$T(w) = NLL(w) + \lambda J(W) \tag{4.3}$$

where J denotes a penalty function for regularization, and coefficient λ controls the severity of punishment to ensure a balance between the models fitness and its complexity.

For the penalty function J selection, L1 regularization would be preferred for its variable selection ability [52]. As a property of L1-norm penalty, if λ becomes sufficiently large, part of the smaller parameters will be forced to zero. On one hand, the variables associated with these smaller parameters (i.e., weaker correlations) may not contribute much useful information in inference; on the other hand, they may actually produce noises and contaminate the inference. By adding L1 penalty to the optimization objective function, the optimization process will force comparatively small values of parameters to go to zero and thus lead to a sparse MRF structure. While for a binary MRF, there is a unique parameter w_c associated with each pairwise clique (i.e., edge), for a general discrete (i.e., multi-class) MRF model, as considered in this application, there is a block of parameters w_c associated with each pairwise clique (i.e., edge). Thus, we need to jointly force all blocks of parameters associated with individual edges to zero. To this end, we employ the so-called Group-Lasso (also referred to as L1L2) regularization method ([52], [71]), and thus have Eq. 4.4

$$J(W) = \sum_{c \in C} (\sum_{i \in c} |w_i|^2)^{1/2} = \sum_{c \in C} \parallel w_c \parallel_2 \tag{4.4}$$

The main advantage of this blockwise regularization is to force all groups of weights to zero simultaneously, so we can achieve sparsity at the block level. When the

objective function is formed as Eq. 4.4, our goal of the optimization process is to compute the block of weights associated to each edge. In this proposal, we define feature functions as event indicators, so the value of an event indicator is either 1 when an event appears, or 0 otherwise. In a pairwise MRF, we have for any edge (s,t), as

$$I_{s,t}(x_i, x_j) = \begin{cases} 1, & x_s = i \text{ and } x_t = j \\ 0, & otherwise \end{cases} \tag{4.5}$$

where i,j represent two states of discrete domain for s, t respectively. One specific event indicator is associated with a corresponding weight $w_(s, t; i, j)$. For instance, in a general discrete pairwise model where x takes values in 0,1,2, each edge is associated with 2×2 possible combinations, and each combination is measured for its importance with a specific weight. Then we have a potential function for each edge as

$$\Psi_{s,t} = \begin{pmatrix} e^{w_{s,t;1,1}} & e^{w_{s,t;1,2}} \\ e^{w_{s,t;2,1}} & e^{w_{s,t;2,2}} \end{pmatrix} \tag{4.6}$$

The exponential family expression of probability over an MRF can greatly benefit the computation. After the weights are learned by the optimization, the values of weights on each edge will determine if an edge exists or not in the optimized target network structure, depending on if the weights of the corresponding block are sufficiently large or not. Since function Eq. 4.4 is non-differentiable, we cannot compute the gradient directly, but, by approximating $J(w)$, we can use the limited-memory BFGS (BroydenFletcherGoldfarbShanno) algorithm to optimize the objective function. In this discussion, our structure learning process starts with the CG. By imposing the Group-Lasso regularization on the CG in the process of information structure optimization, we achieve a new subnetwork topology DG by removing those edges encoded with weaker correlation relationships in the CG provided their removal would not partition the CG. The crucial coefficient λ controls the severity of penalty of Group-Lasso regularization, which in turn controls the level of sparseness of the constructed DG. However, it is difficult to determine λ analytically, due to the fact that

its value is application-dependent to some extent, such as the number of training samples available and the average neighborhood size of the CG.

In practice, the level of sparseness of the constructed DG, i.e., the condition check of line 10 in the above algorithm of DG structure learning, is usually determined by cross-validation, which will be illustrated in the simulation of this section. As also shown in the simulation, the value of λ will be fixed once it is found offline by cross-validation, so λ will not cause a problem from the computation point of view. In other words, the proposed WSN information modeling process will be conducted offline. To quantify the sparseness level of the reduced DG, as compared to the original communication graph CG, we define sparseness ratio (SR) as the ratio of the total number of edges in the DG to the total number of edges in the original CG. A smaller SR value indicates a sparser DG with respect to its original CG. To reduce the complexity of the regularized optimization, we use Pseudo-likelihood to approximate NLL [56], which is proven to be a consistent estimator of the parameters.

4.2 Simulation and Analysis

The data sets for experiment of Data Graph are the same as for the basic model, the in-door temperature measures collected from Intel Berkeley Research Lab and a new out-door WSN collecting humidity measurements from a Redwood in Sonoma California[61]. We also use the same simulation setup for both these two networks. For the convenience of comparison, we select the same 50 nodes from in-door network as the basic model from the the 54 Mica2Dot motes, operating on TinyOS, spreading over the whole lab and the discrete space is built the same way, descretizing the room temperature from 15 to 30 degrees Celsius into 15 discrete states with the constant step as 1 degree. We selected 21 nodes from the Redwood out-door network and we evenly mapped the measurements in rang of [30 70] to 20 discrete states, with size of step=2. For both CG and DG, each node represents a mote (with its temperature sensor), and each edge represents a communication link. Based on CG we build, we

will construct the subgraph DG following the procedure of graph reduction illustrated in Figure 4.3

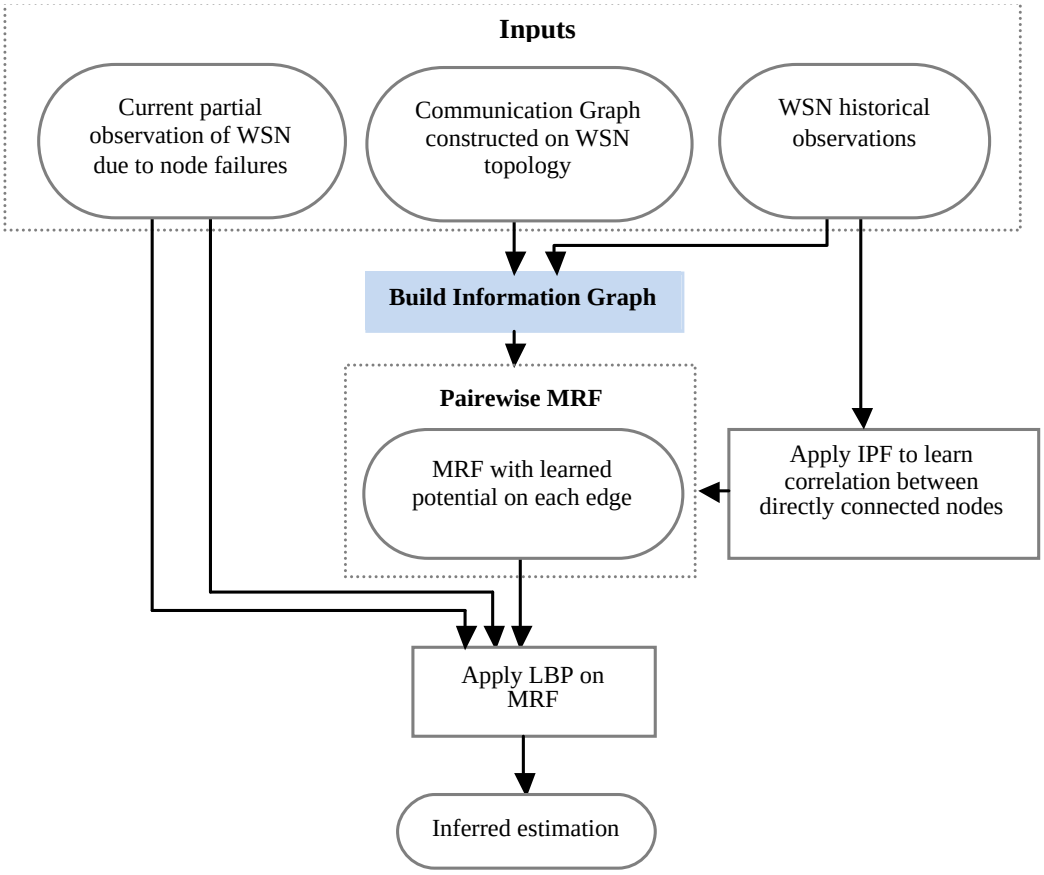


Figure 4.3. Flowchart of simulation of estimation with DG.

To form a robust CG for the Intel Berkeley WSN, we selected four nearest neighbors (i.e., adopted k=log n) to construct each neighborhood to meet the robustness consideration based on [68] and [3]. According to the aggregated connectivity statistics provided by the Intel Berkeley Lab, shorter distances did lead to lower packet dropping rates, justifying the CG model based on k-nearest neighbor graph. Similarly, for the redwood WSN, a robust CG can be formed by selecting three nearest neighbors for neighborhood of each mote.

Starting from the constructed CG, total 80 training sets are available to learn the DG; 10 additional data sets are used for validation; and 10 other data sets are reserved for testing in our simulation. For the purpose of comparison, the training and test samples will also be the same as the Belief Inference in the basic model. The 10 validation data sets are used to select an appropriate λ to construct our optimized DG. For each validation/test case (i.e., test data set), we randomly select a fraction of motes with missing readings to be the estimating targets of the application. For the redwood outdoor WSN, there were 90 training sets available to learn the DG from its robust CG, 20 different data were set aside for validation and the other 20 data for test.

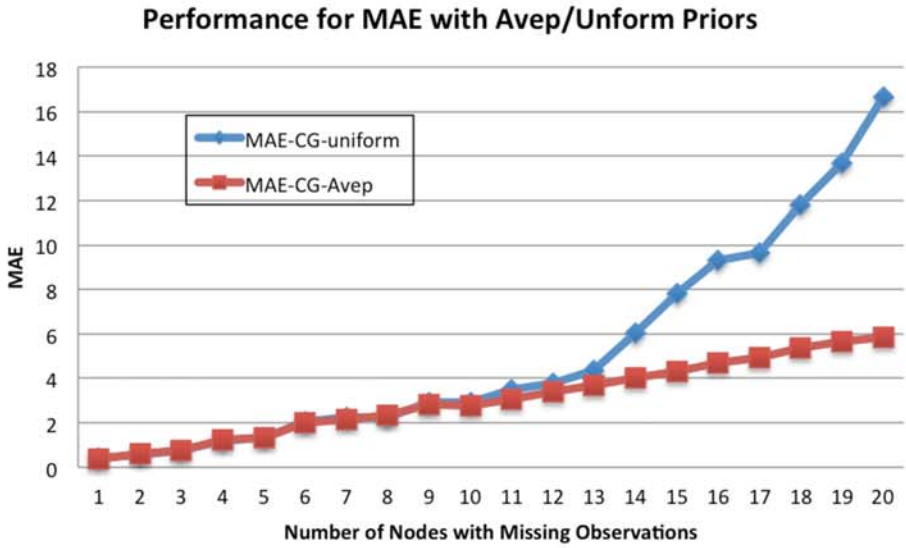


Figure 4.4. MAE comparison between LBP based on Avep and uniform priors.

As a result, we use uniform priors in the simulation in this section for the convenience of computation. When uniform priors include no prior information to the inference process, all the support information will come from the parameter learning process using IPF, in the form of potential function associated to each link. As a counterpart, another target estimation is performed on the DG constructed by our

proposed approach (LBP-DG), with the same LBP inference procedure, and uniform prior configuration (as in LBP-CG). We then evaluate both the estimation performance and energy efficiency of each method. During the estimation process, we gradually increase the number of unobserved nodes and evaluate the performance of LBP-DG with MAE.

4.2.1 Simulation Data and Setup

The indoor sensor network of Intel Berkeley Research Lab consists of 54 Mica2Dot motes, operating on TinyOS, spread over the whole Lab, as illustrated in Figure 4.5. We selected 50 motes with enough temperature readings in our simulation, as the remaining four motes have a significant number of missing measurements. It is reasonable to assume that the room temperature ranges from 15 to 30 degrees Celsius and thus can be discretized into 15 discrete states with the constant step as 1 degree. The outdoor WSN in Sonoma, California collected sensing humidity data (among others) over 33 nodes from a 70-meter tall redwood tree, with sampling rate of 5 minutes. The motes were distributed over the tree with about 2m spacing, 15m to 70m from ground level, 0.1-1m from the trunk, as illustrated in Figures. 4.5 and 4.6. One feature of this redwood WSN is its low packet reception rate, so we selected 21 nodes collecting relatively more data samples for our simulation to avoid the potential effect of contaminated measures. Similarly, we discretize the measurements to 20 discrete states.

In our pairwise MRF DG models for these WSN applications, each node represents a mote (with its temperature sensor), and each edge represents a communication link. To form a robust CG for the Intel Berkeley WSN, we selected the four nearest neighbors (i.e., adopted k=log n) to construct each motes neighborhood to meet the robustness consideration based on [28, 29]. According to the aggregated connectivity statistics provided by the Intel Berkeley Lab, shorter distances did lead to lower packet dropping rates, justifying the CG model based on k-nearest neighbor graph.

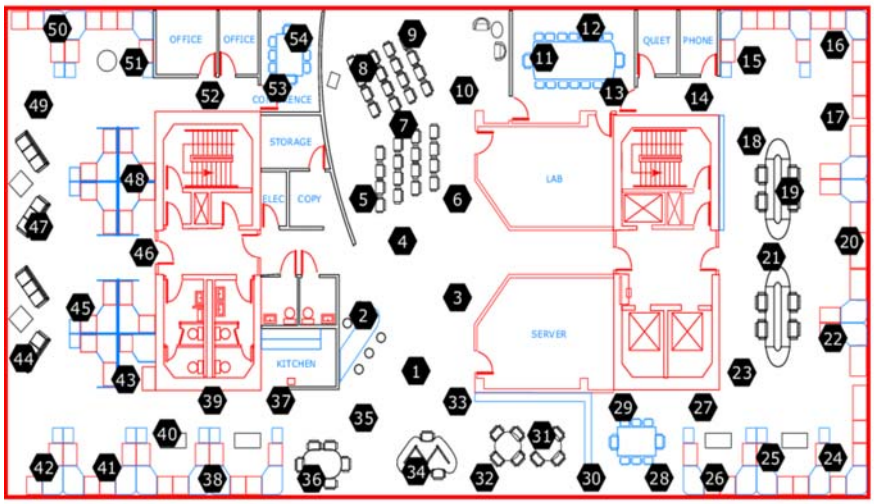


Figure 4.5. Illustration of IntelLab topology.

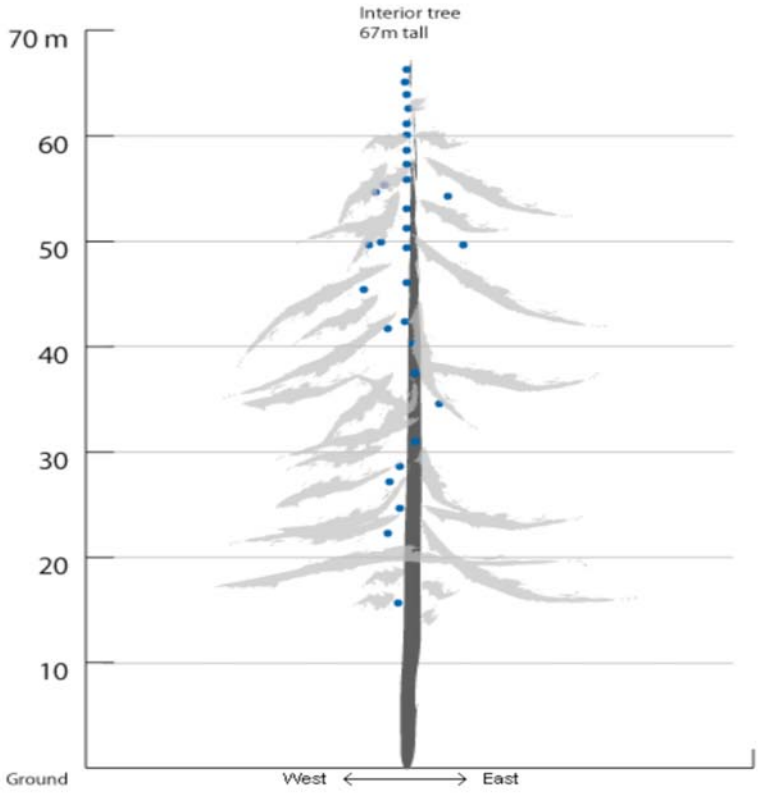


Figure 4.6. Illustration of Redwood topology.

Similarly, for the Redwood WSN, a robust CG can be formed by selecting the three nearest neighbors for each motes neighborhood.

For the Intel Berkeley WSN, starting from the robust CG formed, a total of 80 training sets were used to learn the DG structure; 10 additional data sets were used for validation and 10 other data sets were reserved for testing in our simulation. The 20 validation data sets were used to select an appropriate λ to construct the optimized DG. For the Redwood outdoor WSN, there were 90 training sets available to learn the DG structure from its robust CG; 20 different data sets were reserved for validation and the other 20 data sets for test. For each test case (i.e., test data set), we randomly selected a subset of motes with missing readings to be the estimating targets of the application.

4.2.2 Simulation with Indoor WSN Data

We average the estimation results, for both LBP-CG & DG, over 30 runs, with different random configurations of missing patterns for all validation and test cases. During the validation process, the value of λ is determined in such a way that it ensures a connected DG with the best inference performance on validation data. To thoroughly investigate the potential impact of training data size on the quality of constructed DG, we conducted experiments using 80 training data sets in DG learning process, and validate the learned DG accordingly.

During the validation process, the value of λ was determined in such a way that it ensures a connected DG with the best inference performance on validation data. As illustrated in Section 4.1, λ determines the sparseness level of DG although there is a lack of a theoretical way to determine an optimal λ value. Through the validation process, we started with a small enough λ and increased its value accordingly, until an appropriate lambda value was found; this process is dependent upon applications, as both the number of training samples available and average neighborhood size of the CG can affect the value of λ found.

The validation performances with different DG structures learning from the CG (i.e., different values of λ) are shown in Figure 4.7, along with the model performance of CG. As one can see in Figure 4.7, the number of edges of a DG, indicated as EdgeNum, is reduced with larger λ. The largest λ value (i.e., λ=19) which can still result in a connected DG structure (with 82 edges, in this empirical study of Intel Berkeley WSN), is referred to as threshold λ for the given CG. We can see from Figure 4.7 that, for different λ values, LBP-DG shows almost the same validation performance (i.e., not sensitive to λ) when the number of missing observation motes is less than or equal to 33 from the total 50 motes.

The selection of λ depends on optimization objectives: 1) producing a sparsest connected DG to maximize the energy efficiency of in-network inference; or 2) producing a robust connected DG. To achieve objective (2) above, we make sure that the minimum neighborhood of each mote in the constructed DG is equal to or higher than the upper bound $0.5139logn$ to guarantee network connectivity with probability close to 1 as the size n of WSN increases. For Intel Berkeley Research WSN data of 50 selected nodes, the minimum neighborhood of size 2 is necessary to satisfy this bound. Thus λ=18 is needed for constructing a robust DG. To summarize, according to our proposed information modeling approach, the sparsest DG constructed (with λ=19) has 82 edges while the robust DG constructed (with λ=18) has 87 edges. In contrast, the CG has 132 edges. The topologies of CG and the learned robust DG structure (λ=18) are given in Figs. 4.9(a) and 4.9(b) respectively, in which each mote is indexed by its mote ID. In this case, SR=0.66.

The validation performances over the given validation data sets are shown in Figures 4.7 and 4.8, with different values of λ used for constructing the DGs. As one can see, the number of edges, indicated by EdgeNum, of a DG is reduced with higher value of λ. The largest λ value (such as λ=19) in the figure is the upper bound of the λ which can be employed to still get a connected DG. We refer this upper bound of the λ as the threshold. We can see from the validation results that LBP-DG shows a clear estimation performance advantage over LBP-CG, for different λ values. It is

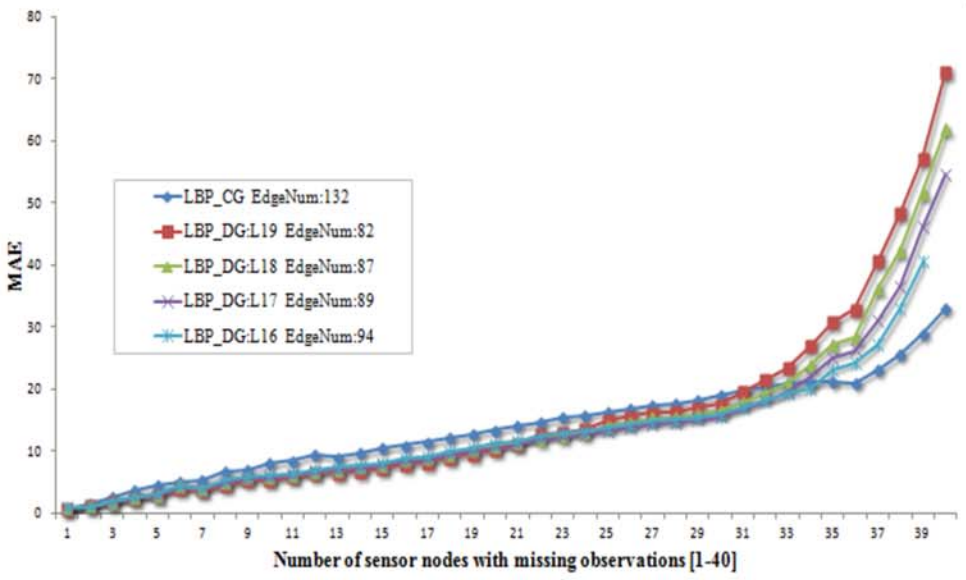


Figure 4.7. Comparison of testing performance of CG and DG with MAE.

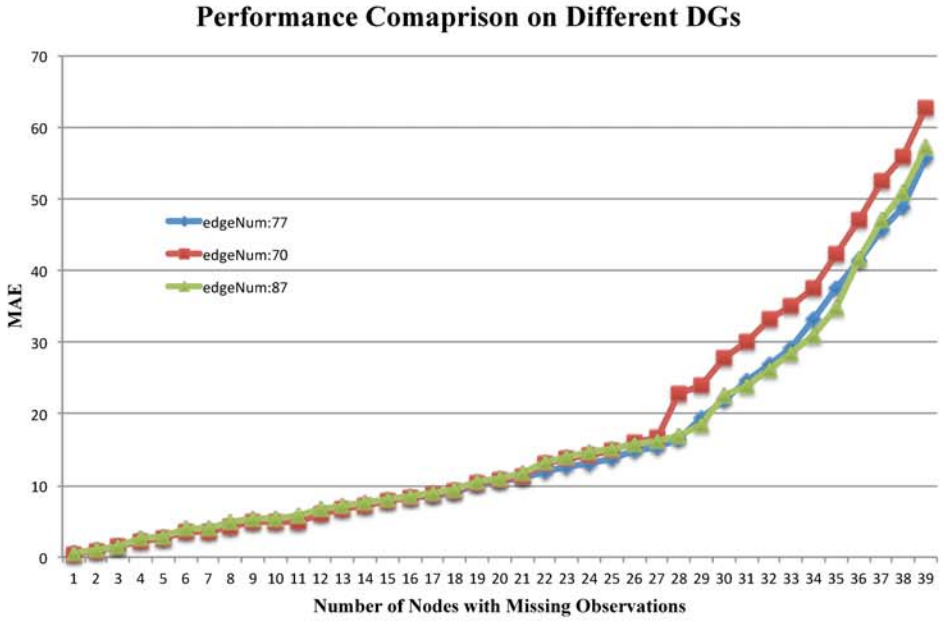


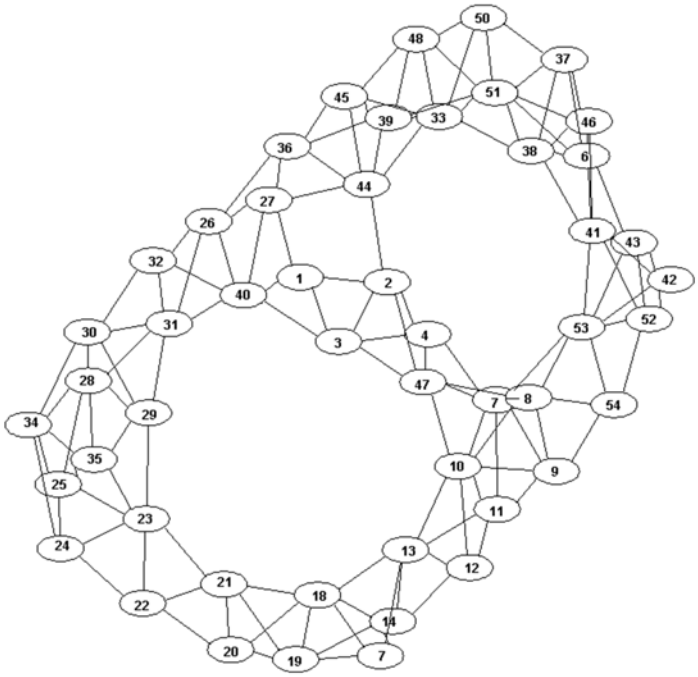
Figure 4.8. Validation performance (training data size: 80).

also easy to see that, the estimation performance is not very sensitive to λ slightly less than the threshold, but there could be some accuracy drop when the value of λ decreases further, as shown in Figure 4.8.

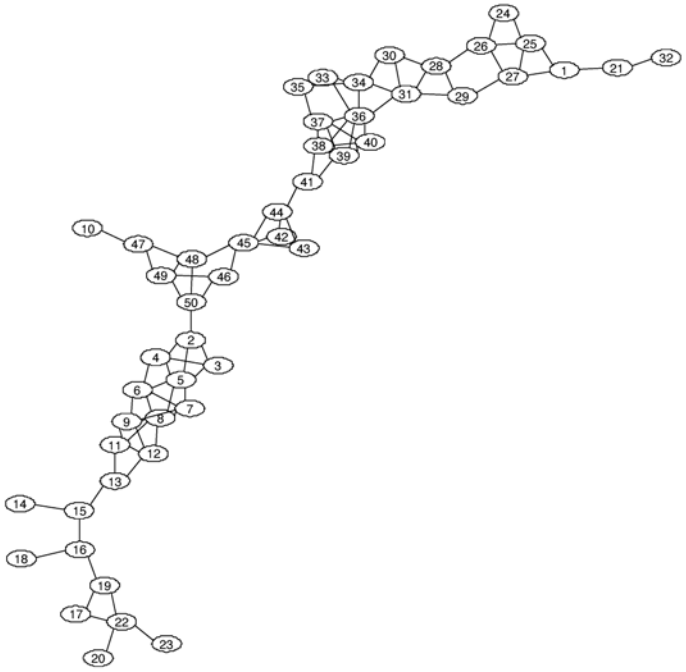
An interesting observation in our empirical study is that the validation performance of LBP-DG degrades with reduced training data size. The DG learned with 60 training samples show no improvement on accuracy even though the reduced graph does provide energy efficiency on communication of the WSN. This observation suggests the amount of training data is probably not sufficient to achieve the best possible DG structure leaning result. The threshold λ also gets smaller with less training samples, dropping from 19 for 80 training samples to 15 for training samples of 60. As the learned DG is not very sensitive to small changes of λ when close to the threshold λ value, we try to select the threshold λ to maximize the reduction on number of edges and maintain a robust connectivity. That is, we choose λ=18 in constructing DG with training data size of 80. The topology of CG and DG is shown in Figure 4.9 for visual understanding of their topological difference.

In the topology graphs, each node represents one sensor mote and is indexed by the real-world sensor ID. we employ the robust DG with 87 edges for evaluation, which can better ensure the no-routing property even in dynamic communication environments. We show the estimation testing performance of the LBP-DG in comparison to that of LBP-CG in Figure 4.10.

We see that, inferring on the reduced network structure DG (34% edge reduction from the CG), the LBP-DG provides a slightly better performance over the LBP-CG when the number of nodes with missing observations are relatively moderate, e.g. less than 50% of the total motes, but shows disadvantage when the percentage of unobserved nodes is very high (i.e., more than 66% of the total motes). Since both LBP-CG and LBP-DG approaches employ the identical LBP inference scheme and use identical partial observations in the distributed in-network inference, the performance gain (when missing readings rate $\leq$ 50%) could only come from the optimized information structure of the data graph from the original WSN connectivity



(a) Topology of the Communication Graph (CG).



(b) Topology of the Data Graph (DG).

Figure 4.9. Comparison between CG and DG topologies.

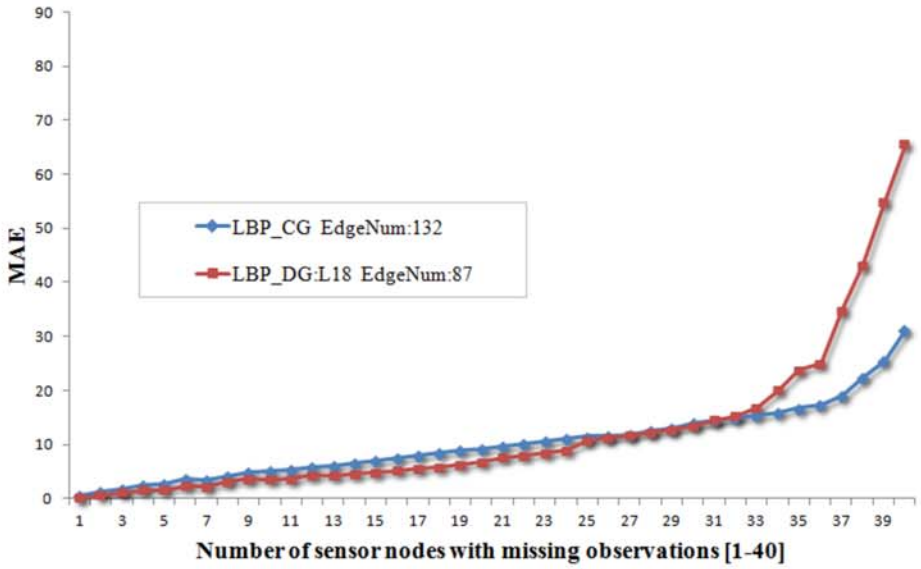


Figure 4.10. Comparison of testing performance of CG and DG on MAE.

graph. Our insight is that, by removing the weak/redundant correlation edges in the CG, the optimized DG actually removes some potential noises introduced by those communication links in the CG but not in the DG and thus leads to slightly better estimation performance. However, when the majority of nodes miss readings, those weak correlation edges in the CG but not in the DG topology could compensate.

To validate our DG-based approach thoroughly, we investigated what estimation performance we can obtain if we use a reduced network topology from a communication perspective rather than from our information modeling perspective to perform in-network inference. An immediate smaller sub-CG alternative to the original CG could be obtained by reducing the sensor network connectivity topology based on geographical distance between motes, where edges representing longer geographical distance in a neighborhood are removed first. This way, a sub-CG (denoted as SCG) is obtained, which is aimed to have a similar topology complexity as the DG. However, the key difference between the DG and the SCG is that the DG is obtained based on the strength of mutual information correlation achieved through our proposed information modeling.

To make a fair comparison with DG, a sub-CG (i.e., SCG) with the same number of edges as DG (i.e., 82) is obtained in our simulation. The resulting topology of the SCG is shown in Figure 4.11 and the corresponding estimation comparison is given in Figure 4.12 .

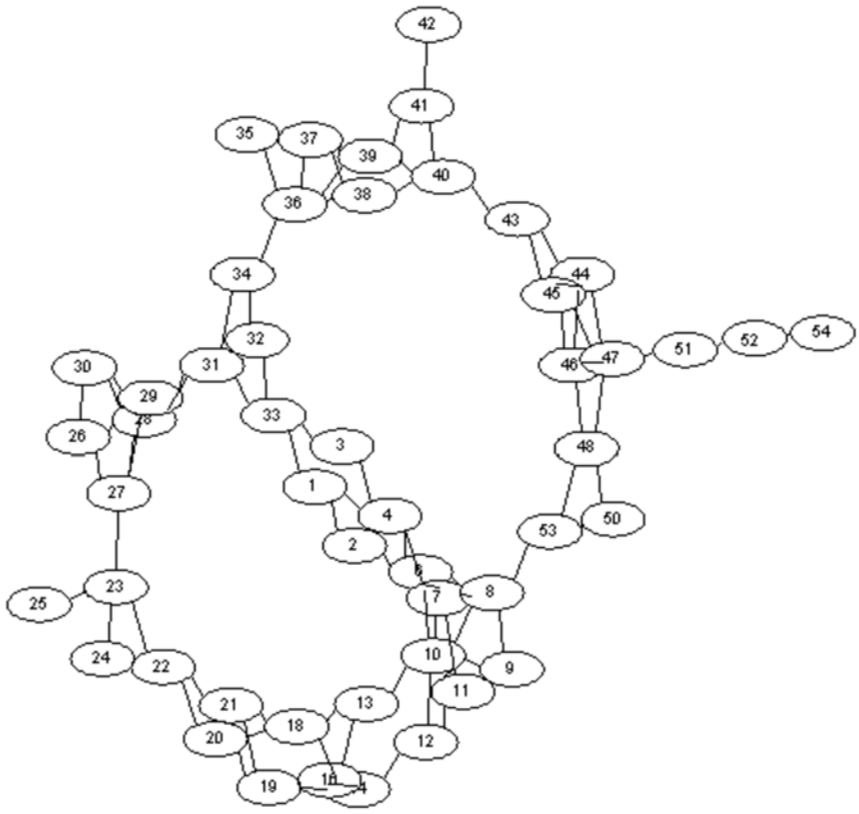


Figure 4.11. Topology reduced based on geographical distance

For the convenience of reference, we call the inference process on the sub-CG topology as LBP-SCG, plotted in red in Figure 4.12. It is easy to see that, with the same number of edges reduced as DG, sub-CG results in a much worse estimation performance compared to that of LBP-CG. On the other hand, the LBP-DG, indicated by green line, show much better performance than both LBP-CG and LBP-DR. In contrast, the approach of sub-CG takes a different direction (i.e., communication perspective) from our information structure optimization approach of DG, although

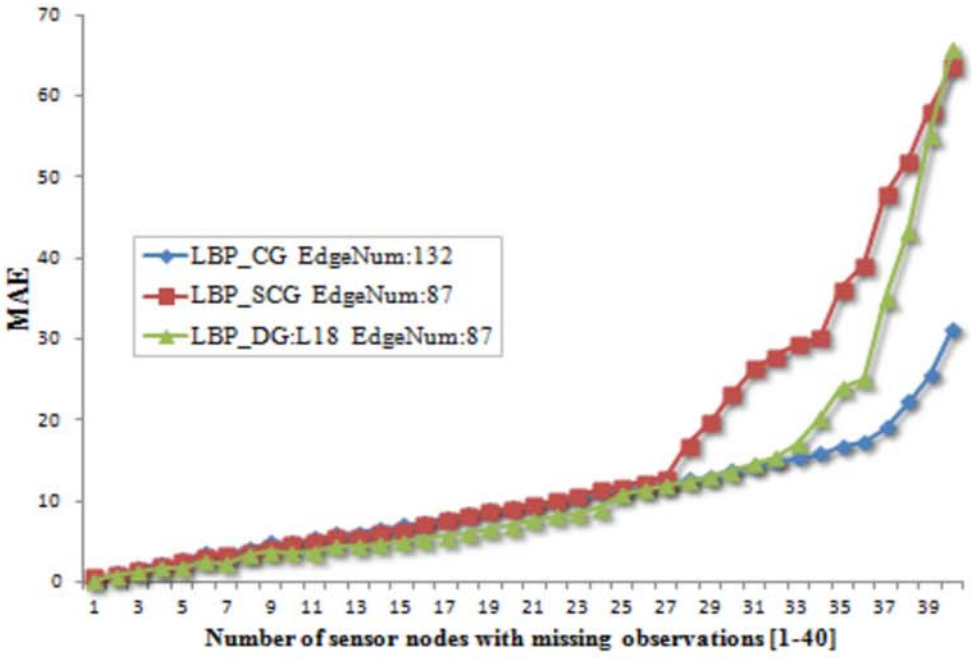


Figure 4.12. Comparison of estimation accuracy.

both approaches would lead to reduced network topologies. One can also observe from Figure 4.9 and Figure 4.11 that sub-CG contains many more short loops than DG, which causes LBP-based inference to have more difficulty to converge.

In summary, we demonstrate that the estimation performance gain achieved by our proposed approach LBP-DG is not due to the reduction of the communication structure of the sensor network, but is due to the optimization of information structure in the sensor network, upon which BP-based inference is conducted. This demonstrates the merit of the proposed approach.

Now, let us consider the energy consumption aspect of in-network inference. In addition to the performance improvement by our DG-based approach for distributed inference in WSN, our approach also significantly reduces energy consumption of the WSN at the same time, due to the dramatically reduced topology of DG from the original CG of the WSN. In the modified LBP for WSNs [5], one broadcast operation from one node can allow all of its neighbors to get new messages. As a basic principle of WSNs, an important motivation behind our proposed method is energy efficiency,

which is mainly affected by the need of message transmission and reception for in-network BP. As a basis for further discussion, we assume: 1) a perfect communication channel (no collision, no packet dropping); and 2) the best energy efficient scenario for BP in CG, that is a node will broadcast only when it has collected messages from its whole neighborhood. We evaluate the energy consumption in terms of counts of transmission and reception operations of the whole network, for both CG- and DG-based inference. Corresponding to the increased percentage of missing nodes, on the Y axis, counts of transmission or reception are averaged over all test cases with 30 runs for each test case and a total of 300 runs. As illustrated in Figure 4.13 , the number of reception operations in DG is significantly reduced from CG (almost 50% lower), due to the reduced size of the neighborhood in DG. That is, the fact that each node needs to collect fewer messages to include all the update information from its neighbors result in Eq. 4.7

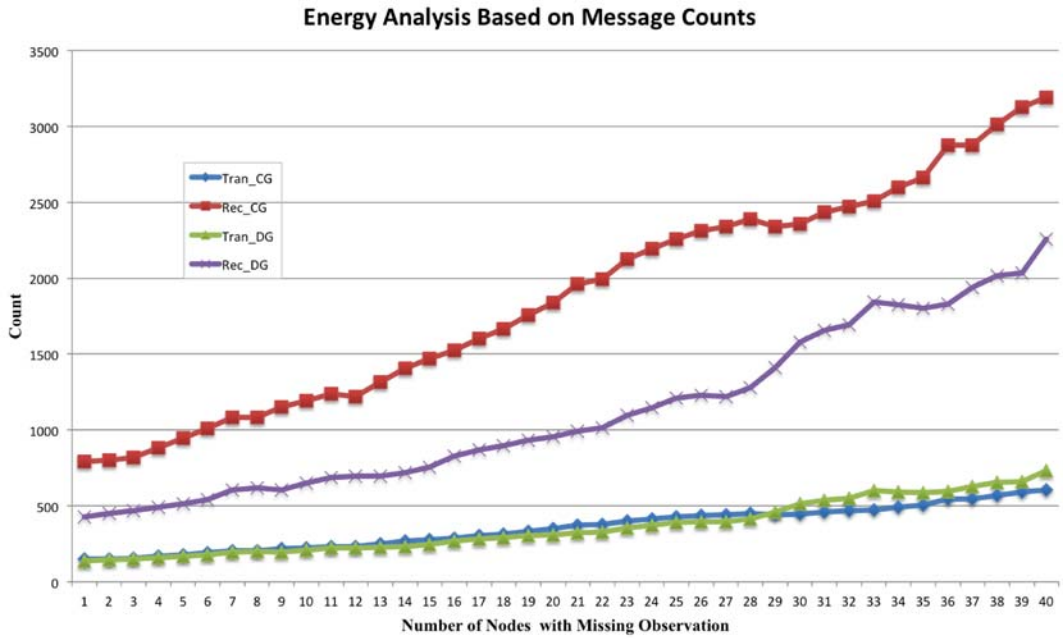


Figure 4.13. Transmission and reception counts.

$$C(Rec - DG) < C(Rec - CG) \tag{4.7}$$

where function C() counts the number of operations. On the other hand, with the first assumption above, the reduced size of the neighborhood in DG will not directly lead to fewer transmission operations (Tran-DG vs. Tran-CG), but only through the impact on the speed of convergence of inference process. Our insight is that in the constructed DG, by Group-Lasso regularization, noisy information is significantly excluded from the inference process and, moreover, DG has many fewer short cycles than the CG so the convergence of inference process will speed up. As a result, DG leads to fewer transmission operations as illustrated in Figure 4.13. We have

$$C(Tran - DG) \leq C(Tran - CG) \tag{4.8}$$

The difference between C(Tran-DG) and C(Tran-CG) increases when more readings are missing as shown in Figure 4.13. With a higher percentage of missing readings, the prior information is further reduced and the inference process needs to take more iteration to converge, if it still can. At the same time, the gap between C(Rec-DG) and C(Rec-CG) is also broadening, because, with the larger neighborhood, C(Rec-CG) increases faster even with the same speed drop. Considering Eq. 4.7 and Eq. 4.8 together, BP inference in WSN based on DG can significantly reduce the total counts of communication operations, both sending and receiving, than the inference based on CG.

4.2.3 Simulation with Outdoor WSN Data

For the redwood WSN, the robust CG formed based on k-nearest neighbor graph connectivity model has total 40 edges. With the same validation procedure as described above for the Intel Berkeley WSN, the value of λ is found to be 12 to maintain a robust connected DG in which the minimum neighborhood of each mote is 2 (i.e., $2 > 0.5139 log 21$) given the redwood WSN size of 21. The topologies of the CG and

the constructed robust DG (with 28 edges) of redwood WSN are illustrated in Figures 4.14 and 4.15, respectively.

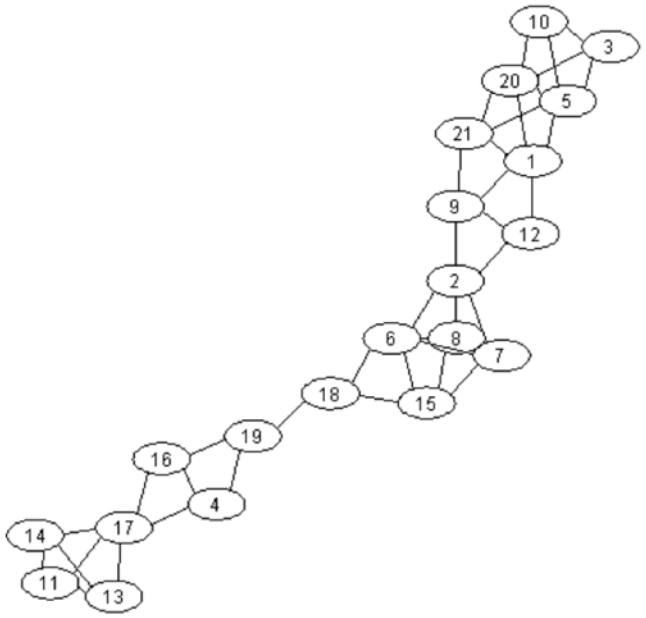


Figure 4.14. Topology of CG for the Redwood WSN.

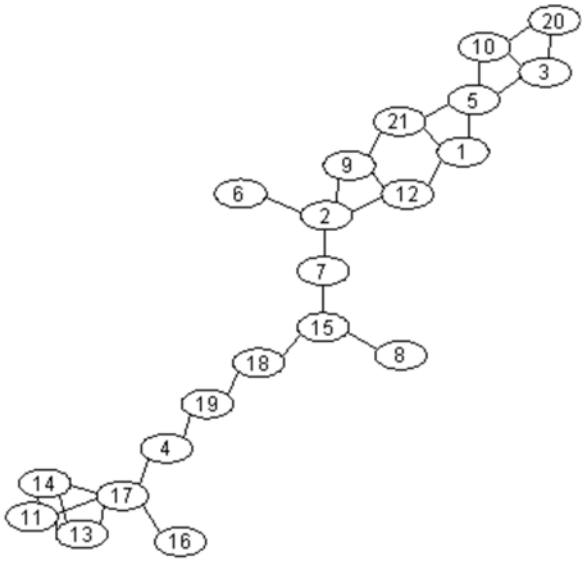


Figure 4.15. Topology of DG for the Redwood WSN (EdgeNum28).

The robustness of DG is just part of criterion that needs to meet. The constructed DG also need to have comparable performance as CG. Through the validation process, the performances of DGs with different number of edges/value of λ are illustrated in the validation process below.

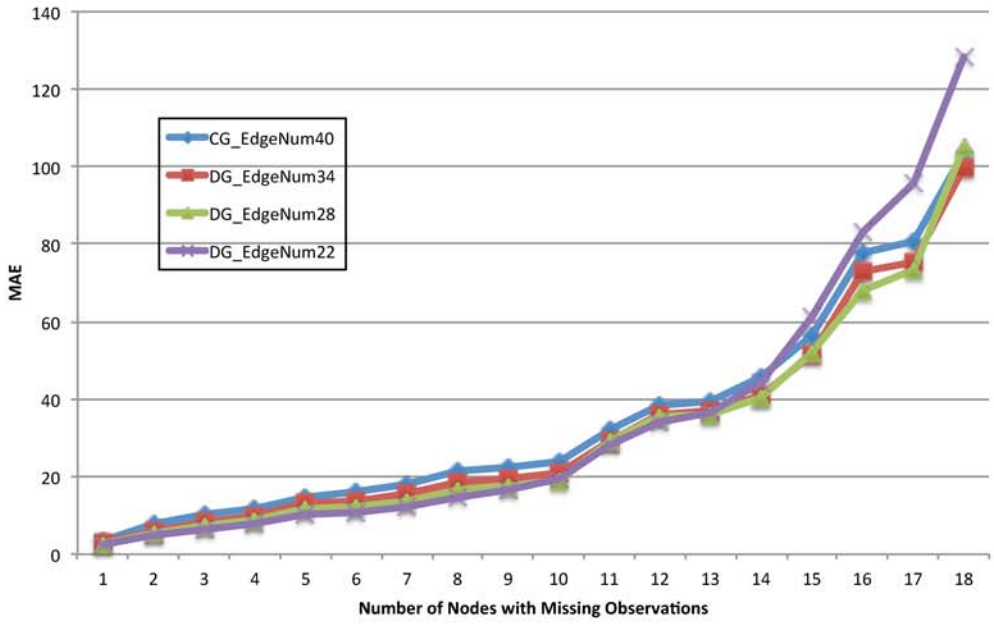


Figure 4.16. Validation performance different lambda/number of edges.

We can see that the DG with 28 edges have similar and even better performance than CG. Although DG with 34 edges also provide similar performance, smaller network in term of edges will always be the choice for energy efficiency purpose. The performance degrades fast when the number of edges drops to 22 even the edges are removed in consistent speed (i.e.6 each time). We conducted our simulation study with LBP on CG and robust DG with 28 edges separately for redwood WSN and the performance results based on MAE are given in Figure 4.17. We can see that LBP-DG approach can achieve the similar or better performance with 30% fewer edges than the CG. The similar analysis in term of message count is also conducted on DG as shown in Figure 4.18

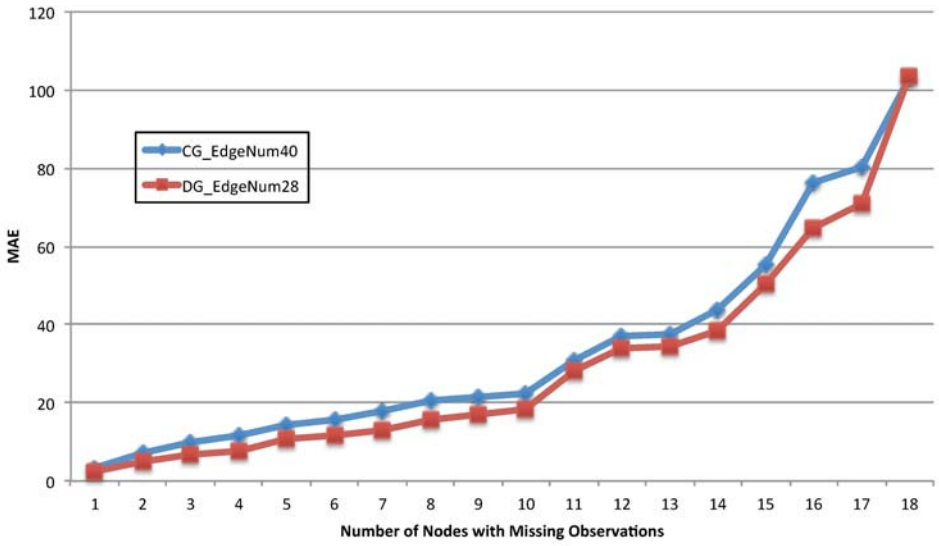


Figure 4.17. Estimation performance comparison of CG and DG (EdgeNum28).

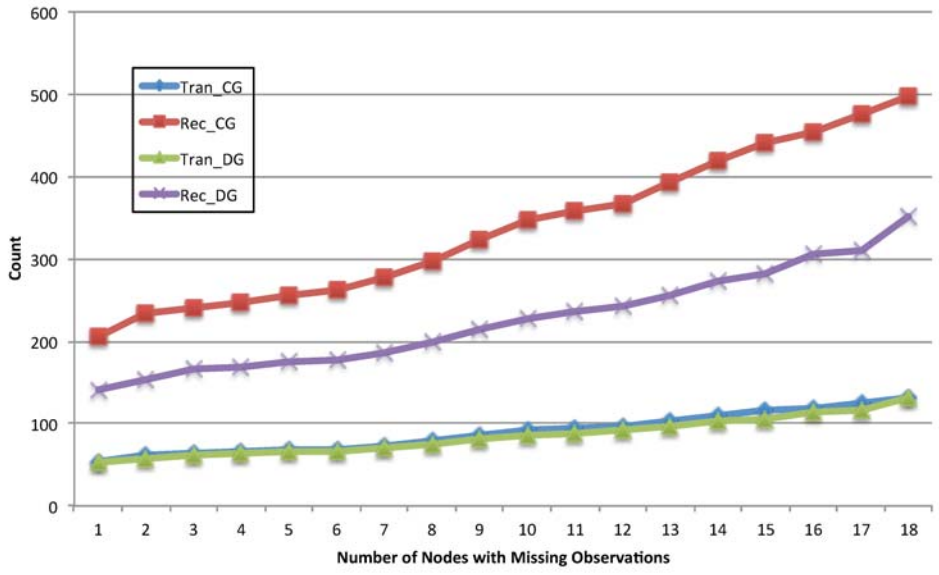


Figure 4.18. Energy analysis based on message count.

5 MULTI-RESOLUTION INFERENCE

One of the most important issues with a sensor network is energy efficiency especially for a large-scale sensor network that can break down because of bottle neck of energy consumption of particular group of sensors. As we know, the major part of energy consumption comes from transmission (i.e. operation of sending/receiving message), we need to address this problem carefully since Belief Inference is a message-propagation based method. This problem is addressed by applying idea of multi-resolution inference based on wavelet transformation and structure optimization. First, a version of Multi-Resolution based only on wavelet transformation of belief message is introduced. Then on top of this model, an advanced version involving DG introduced in Section 4.1 is proposed with more flexibility and better performance. Since the computation complexity of the advanced Multi-Resolution model will increase dramatically with the size of the target network, the simpler version is still necessary to handle large-scale sensor network (e.g. network with 1024 nodes in our simulation).

5.1 Wavelet Based Belief Propagation

5.1.1 Methodology

Wavelet based belief propagation, W-LBP, takes advantage not only the space correlation of adjacent sensor nodes, but also the natural multi-resolution property of wavelet, so it will be easier to go through the basic idea of wavelet theory first. Wavelet theory provides a mathematical tool for hierarchically decomposing signals. Mathematically, the mother wavelet function satisfies

$$\int_{-\infty}^{\infty} \Psi(t)dt = 0 \tag{5.1}$$

The wavelet basis functions which project the original signal to a wavelet coefficient domain are achieved by scale and shift operation on the mother wavelet function.

$$\Psi_{s,\tau}(t) = \frac{1}{\sqrt{s}}\Psi(\frac{t-\tau}{s}) \tag{5.2}$$

where and denote shift and scale factors, respectively. The one-dimensional (1-D) wavelet transformation is actually the inner production of signal and as

$$W_f(s,\tau) = \int_{-\infty}^{\infty} f(t)\Psi_{s,\tau}(t)dt = \langle f, \Psi_{s,\tau}\rangle \tag{5.3}$$

The discrete wavelet transformation (DWT) was developed to apply the wavelet transform to digital signals. Mallat introduced a tree algorithm for computing DWT by using filter banks [43], in which any original digital signal is decomposed into the approximated signal and the corresponding detail signal through low-pass (h) and high-pass filters (g), related as quadrature mirror filters. Since each filter halves the frequencies of the signal, the filter outputs are subsampled by 2. For one level decomposition, the transform coefficients, a_k and d_k, have the following expression:

$$\begin{aligned} a_k^{j-1} &= \sum_n h_{n-2k} a_n^j \\ d_k^{j-1} &= \sum_n g_{n-2k} a_n^j \end{aligned} \tag{5.4}$$

where j denotes the resolution and k is the index for the samples. For single level signal decomposing and reconstructing, it can be illustrated in Figure 5.1. In Figure 5.1, high-pass and low-pass analysis filters, indicated with dotted squares, are denoted as H and L respectively, whereas the corresponding synthesis filters in the reconstruction process are denoted as H^* and L^* (the transposed matrix of H and L, respectively). In fact, this decomposition process can be applied recursively to the approximate coefficients until the desired result is reached. The finest resolution level is the original signal [62].

In the case of discrete random variables, the belief message is a vector of numbers. Our idea is to adopt wavelet methodology to compress the belief message by dropping

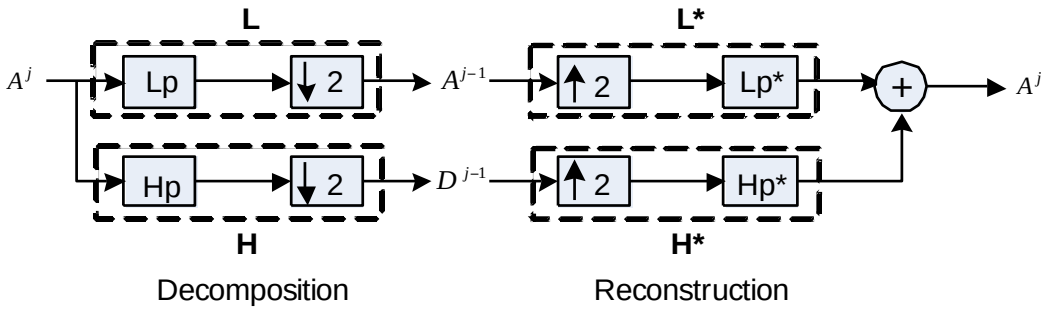


Figure 5.1. Decomposition and reconstruction.

its details at the sender site, and thus only to transmit the approximation of the original belief message to its one-hop neighbor. On the receiver side, the details of the local belief are then used to reconstruct and estimate the original belief message, before further operation with potential function. In W-LBP, the message transmitted from site i to j is

$$\omega_{ij} \leftarrow \sum_{x_i} \varphi_i(x_i) \prod_{k \in N(i)/j} \psi_{kj}(x_k, x_j) \hat{\omega}_{kj}(x_i) \tag{5.5}$$

Accordingly, the expression for local belief will be

$$b_i(x_i) = K \varphi_i(x_i) \prod_{j \in N(i)} \psi_{ji}(x_j, x_i) \omega_{ji}(x_i) \tag{5.6}$$

To decompose at site i, as shown in Figure 5.2, we have

$$A_{ij} = L\omega_{ij} \tag{5.7}$$

$$D_i = Hb_i \tag{5.8}$$

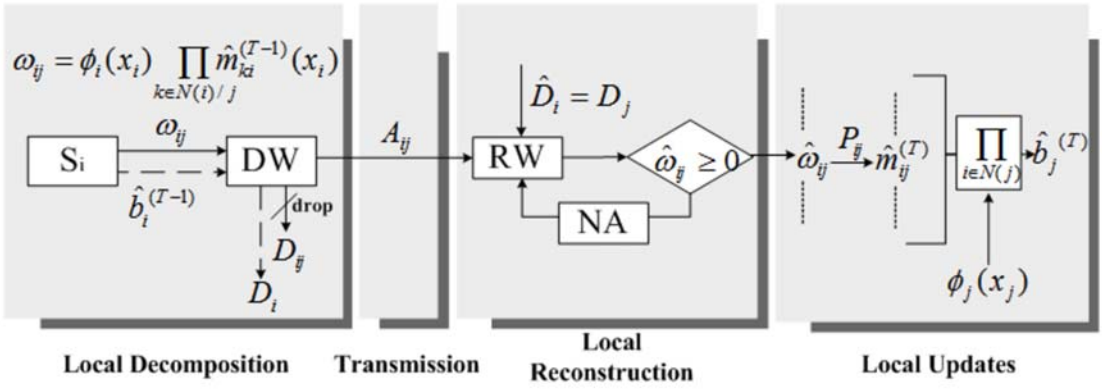
As original D_{ij} is not available at site j, to reconstruct and estimate ω_{ij} at site j, we have

$$\hat{D}_{ij} = D_j \tag{5.9}$$

$$\hat{\omega}_{ij} = L^* A_{ij} + H^* \hat{D}_{ij} \tag{5.10}$$

Although one level wavelet decomposition is depicted in Figure 5.1, multilevel wavelet decomposition may be preferred in some situations due to severe energy limitation, as the size of an approximated message with DWT will be reduced to 2^{-m} of the original message size, where indicates the level of wavelet decomposition. To reduce wavelet operations at sensor nodes, Haar wavelet is adopted in our experiment, which is defined as step functions:

$$\Psi(t) = \begin{cases} 1 \ 0 \leq t < 1/2 \\ -11/2 \leq t < 1 \end{cases} \tag{5.11}$$



$$S_i : Site\ i \tag{5.12}$$

Figure 5.2. W-LBP process.

It is important to note that, as a natural property of belief distribution, there should not appear any negative element at any moment during belief inference. This property can be exploited to improve our estimation on the dropped details at site i during the reconstruction process on receiver site j. Whenever any negative element occurs in the estimated $\hat{\omega}_{ij}$ by Eq. 5.10, due to the inappropriate $\hat{D}_{ij}$, some adjusting operations are conducted and get a better estimation of $\hat{D}_{ij}$, so that the previous

negative elements in $\hat{\omega}_{ij}$ become zeros after adjustment. The adjusting operations depend on the type of wavelet employed in W-LBP. We have:

$$\hat{\omega}_{ij} = \{s_n\}_{n=1:N} \tag{5.13}$$

$$\hat{D}_{ij} = \{d_k\}_{k=1:N/2} \tag{5.14}$$

$$\hat{D}'_{ij} = \{d'_k\}_{k=1:N/2} \tag{5.15}$$

For Haar, we have

$$d_k = \begin{cases} d_k - \frac{s_i}{\sqrt{2}}, & s_i < 0 \ and \ i = 2k-1 \\ d_k + \frac{s_i}{\sqrt{2}}, & s_i < 0 \ and \ i = 2k \end{cases} \tag{5.16}$$

We then obtain the improved as follows

$$\hat{\omega}_{ij} = L^* A_{ij} + H^* \hat{D}'_{ij} \tag{5.17}$$

During wavelet transform of a signal, the approximation part always conveys the most important information and thus is kept untouched during the process of estimation. On the other hand, due to the missing of the original details, it can be expected that the accuracy of the estimation using W-LBP would not be as good as that using traditional LBP. The motivation for such estimation is that LBP requires massive data communication which is very energy consuming in a wireless network. Studies have indicated that about 3000 instructions could be executed for the same energy cost as sending a bit for 100 meters by radio [51]. For one level Haar transformation and corresponding estimation, the computation is very simple, which only need a few additions and multiplications as shown in Eq. 5.5, Eq. 5.6. The energy consumption for such low workload on the processor is negligible compared with the energy savings of the reduced communications.

5.1.2 Simulation and Analysis

We studied the application of missing observations estimation with W-LBP in environmental monitoring using lattice real-world soil moisture sensing data from the

Southern Great Plains Hydrology experiment of 1997 (SGP97) in Oklahoma, the same data set deployed for the basic estimation procedure for the performance comparison. The main simulation procedure is illustrated in Figure 5.3, in the similar pattern as basic probabilistic estimation for better understanding. To simulate the missing readings, we randomly designated a certain percentage of broken sites, increasing from 5% to 50% out of a total of 1024 sites. Such partial observations for the MRF model come from two test days, July 14, the dry day and July16, the wet day. For each site with missing readings, an agent among its one-hop neighbors was selected to perform distributed fusion for the missing site. Agent selection protocol was out of the scope of our discussion and will be discussed in future research. In the LBP/W-LBP process, site readings were used as the initial priors for those working sites, while for each missing site, the average value over all local beliefs from its direct neighbors (one-hop neighbors) was obtained as its initial prior.

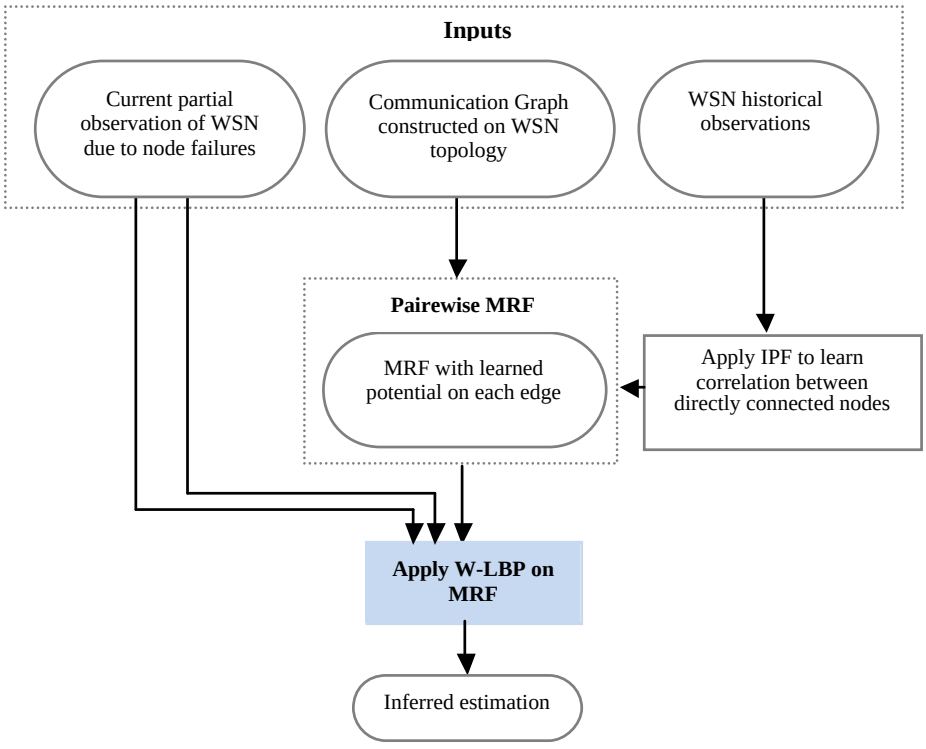


Figure 5.3. Flowchart of simulation of estimation with W-LBP.

The purpose of distribution fusion based on LBP/W-LBP on MRF is to handle the uncertainty problem through fusing the information from partial observation and the correlation information encoded in each potential associated with edges of an MRF model. The statistical relationships (i.e., spatial correlations) are embedded in historical readings and thus can be obtained through a learning process. As shown in Figure 5.3, the communication load of W-LBP is reduced to $1/2^n$ of original LBP, where n is the level of signal decomposition. If only one level of wavelet decomposition is employed in the W-LBP process, such as in this empirical study, then 50% energy conservation compared to the original LBP can be achieved. However, such energy savings come at the expense of some minor degradation of estimation performance. To understand the tradeoff between the energy conservation and estimation performance offered by W-LBP, we conducted comparison experiments between traditional LBP and our proposed W-LBP to infer missing observations given identical partial observations on the same MRF constructed via IPF.

For both test cases, 20 trails of distributed inference on each different percentage of randomly assigned missing sites were performed, and the average performance on inference accuracy is reported in Figure 5.4 and Figure 5.5 for dry day test data and wet day test data, respectively. Two observations can be obtained: 1) In general, there is only a slight degradation on the estimation performance of W-LBP compared to traditional LBP, which is under 5% for the dry day and 9% for the wet day; 2) W-LBP inherits the robustness property from LBP: the accuracy rate decreases gradually as the percentage of missing reading sites increases, and there is no sudden performance drop even when half of the monitoring sites are missing (either broken or in sleep).

Furthermore, in addition to the comparison of accuracy rates, we also analyzed the inference errors in our experiments when accurate estimation was not achieved with LBP or W-LBP. As listed in the tables below, estimation error severity has been classified into three levels: level one indicating the inference error bounded by discrete state, level two indicating the inference error bounded by number of discrete states, and level three indicating the inference error bounded by number of discrete

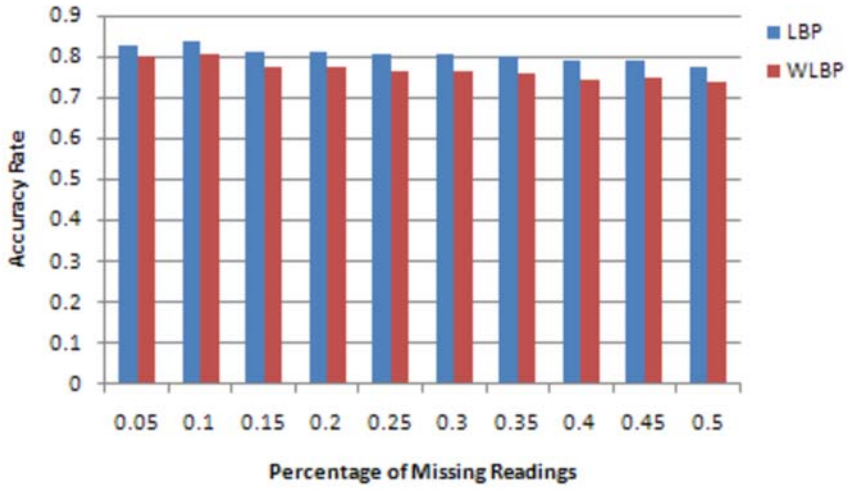


Figure 5.4. Accuracy comparison between W-LBP and LBP (dry).

states, respectively. As we can see, more than 96% and 93% of estimation errors with W-LBP fall into level one even when the missing readings exceed 50% for dry day test cases and wet day test cases, respectively. Overall, the error severity distribution achieved with W-LBP is comparable with that achieved with traditional LBP.

Note that the small fluctuation in the error distribution with the percentage of missing sites is due to the different geographical distributions of missing sites randomly selected in the grid in our experimental trials. The spatial distribution of estimation errors by W-LBP and LBP are illustrated, respectively, on wet day test cases with the same random distribution of an initial 50% missing readings. As we can see, W-LBP actually produces an estimation error pattern very close to that which results from traditional LBP except for several more level 2 and level 3 errors. These results indicate that, even with substantial missing observations, W-LBP inference still inherits robustness property from LBP regarding the accuracy rate and error bounds, but with dramatic energy savings.

//

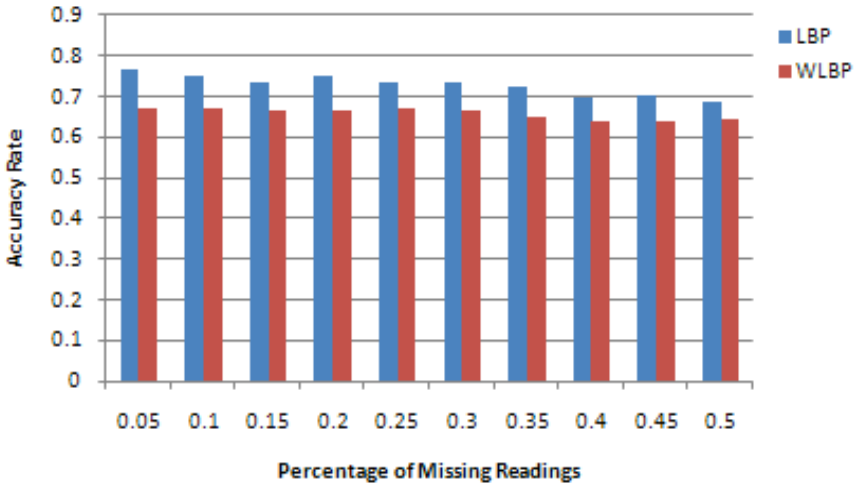


Figure 5.5. Accuracy comparison between W-LBP and LBP (wet).

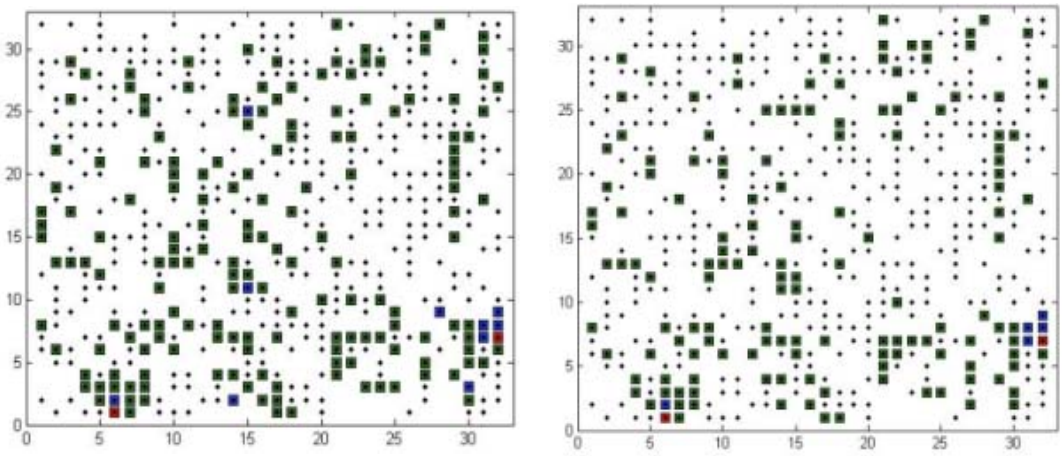


Figure 5.6. Geographical distribution of errors of LBP. The black dots represent the positions of missing readings, and, on top of them, green, blue or red square indicates error level one, two or three. All the rest positions are the sites correctly estimated.

Table 5.1.
Error severity distribution (dry day; 20 trials)

	LBP			W-LBP		
MissingNodes (%)	1	2	3	1	2	3
25%	98.8%	1.11%	0	97.59%	2.40%	0
30%	99.17%	0.83%	0	97.38%	2.54%	0.07%
35%	98.89%	1.11%	0	97.31%	2.68%	0
40%	98.97%	1.03%	0	97.69%	2.25%	0.05%
45%	98.52%	1.48%	0	97.01%	2.98%	0
50%	97.85%	2.10%	0.04%	96.97%	2.91%	0.11%

Table 5.2.
Error severity distribution (wet day; 20 trials)

	LBP			W-LBP		
MissingNodes (%)	1	2	3	1	2	3
25%	97.77%	2.23%	0	93.14%	6.86%	0
30%	97.78%	2.22%	0	94.01%	5.99%	0
35%	97.43%	2.57%	0	94.66%	5.30%	0.04%
40%	96.75%	3.25%	0	94.15%	5.85%	0
45%	96.44%	3.49%	0.07%	93.65%	6.14%	0.21%
50%	95.85%	4.03%	0.13%	93.25%	6.42%	0.33%

5.2 Multi-Resolution Inference: Based on Data Graph

In this section, we investigate the multiresolution inference in which multiple resolutions of structures are combined with the message-passing inference based on our W-LBP method. In this advanced multiresolution inference, the original belief messages were propagated on the edges of DG, whereas the approximations of the

original belief messages were propagated on those edges in the CG but not in the DG. The idea is illustrated in Figure 5.7 in comparison with basic model. The rationale is that since all links in the DG represent stronger correlations, thus full original belief messages are exchanged over DG. On the other hand, since those links in the CG but not in the DG represent weaker correlations, thus only the approximation of the original belief messages exchanged to reduce communications and save energy. In Figure 5.8, the idea of multi-resolution based on DG is illustrated with the same example used in Section 4. Each red dotted line indicates an edge outside DG, which is considered to contain weak correlation, and wavelet transformation will be performed on messages along those edges to minimize the loss of valuable information with the same energy saving on the communication.

When the number of edges further decreases with larger λ value, the performance of DG will degrade, especially with more percentage of missing nodes present. It is the scenario that multi-resolution inference can help. Comparing to DG building process, the connectivity constraint can be ignored to form a sparse network for DG based multi-resolution inference since CG can always be activated for communication purpose. Following the similar process as discussed in Chapter 4, we can get more sparse structure using higher lambda as shown in Figure 5.9.

With such sparse DG illustrated in Figure 5.9, the performance comparisons of multi-resolution inference, labeled as WLBP_CG_DG, on CG and DG, are illustrated in Figure 5.10 for indoor sensor network, sensor network deployed in Intel Berkeley Laboratory. For IntelLab data, there is performance gap between CG and DG when higher percentage (i.e.$>$ 56%) of nodes are missing, presenting the need for multi-resolution inference. The rational is that more information is required to recovery higher percentage of missing observation so, with extra information, the approximation part of messages passed along edges outside DG, WLBP_CG_DG shows its advantage over DG and has better robustness. Another important aspect to exam is energy-efficiency of WLBP_CG_DG, in comparison to CG, which is illustrated in Figure 5.11 and Figure 5.12. As shown in Figure 5.11, WLBP_CG_DG reduces the

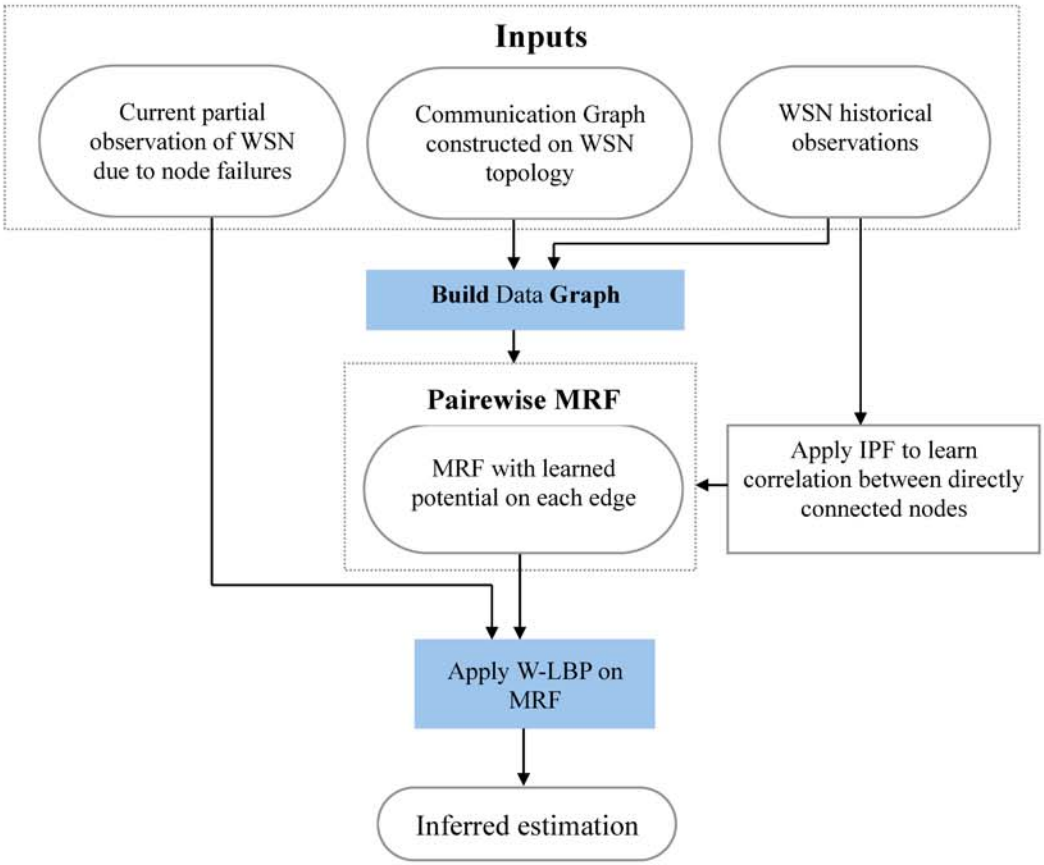


Figure 5.7. Flowchart of simulation with Data Graph based Multi-Resolution Inference.

number of receiving messages in average for each node when transmit almost the same amount of message as CG. Such improvement of energy efficiency was contributed by the one level wavelet decomposition on messages passed along edges outside DG. When comparing to DG in term of energy efficiency, we expect higher energy consumption with WLBP_CG_DG since it, by theorem of multi-resolution inference, balances between energy saving, addressed best by DG and performance robustness, achieved by CG. As demonstrated by Figure 5.12, although DG has better overall energy saving, such advantage over WLBP_CG_DG tend to diminish with higher percentage of missing observations, in term of both count of messages received and transmitted.

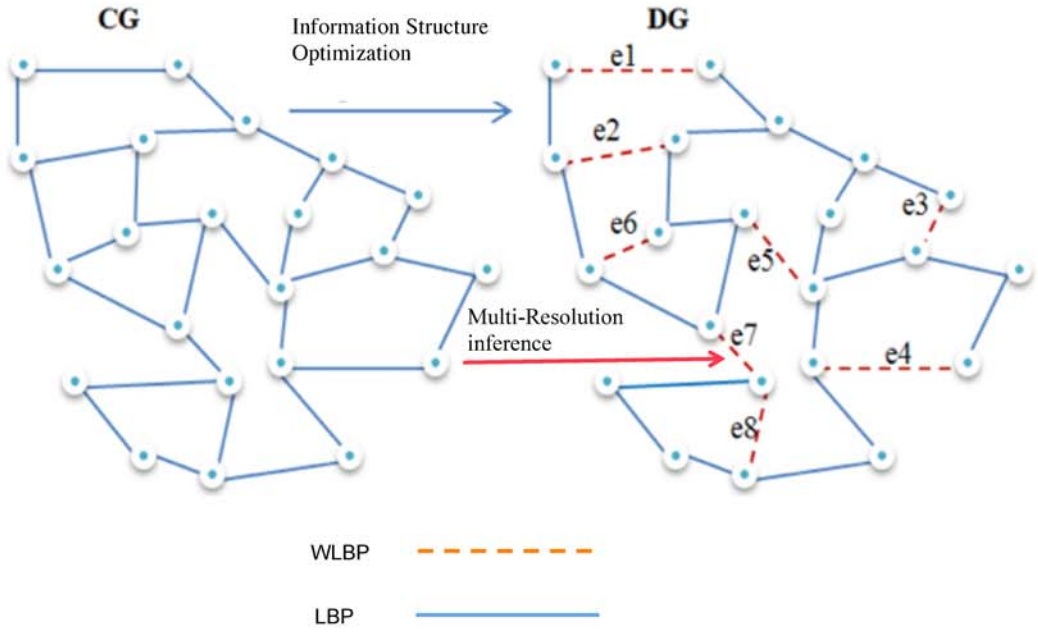


Figure 5.8. Demonstration of DG based Multi-Resolution Inference.

The same analysis on performance and energy saving properties will also be performed on data collected from outdoor sensor network, the redwood case. The more sparse DG was built as shown in Figure 5.13. Based on DG in Figure 5.13, the performance of WLBP_CG_DG with outdoor data was analyzed as shown in Figure 5.14, which follows the similar pattern as performance comparison with indoor data. The overall performance of WLBP_CG_DG is similar as CG after dropping detail information carried by messages along edges outside DG. The analysis of energy efficiency was illustrated in Figure 5.15.

Interestingly, the multiresolution inference with W-LBP achieved the similar estimation performance to the best performance either on CG or DG, particularly when missing observation is severe (i.e. missing rate higher than 67%) in the redwood WSN, and at the same time, the multiresolution inference is obviously more energy efficient than the CG-based approach. We note that the MAC layer communications of WSN need to be modified accordingly to achieve this energy efficiency in multiresolution inference. Basically, every mote sends its belief message in a wavelet-decomposed

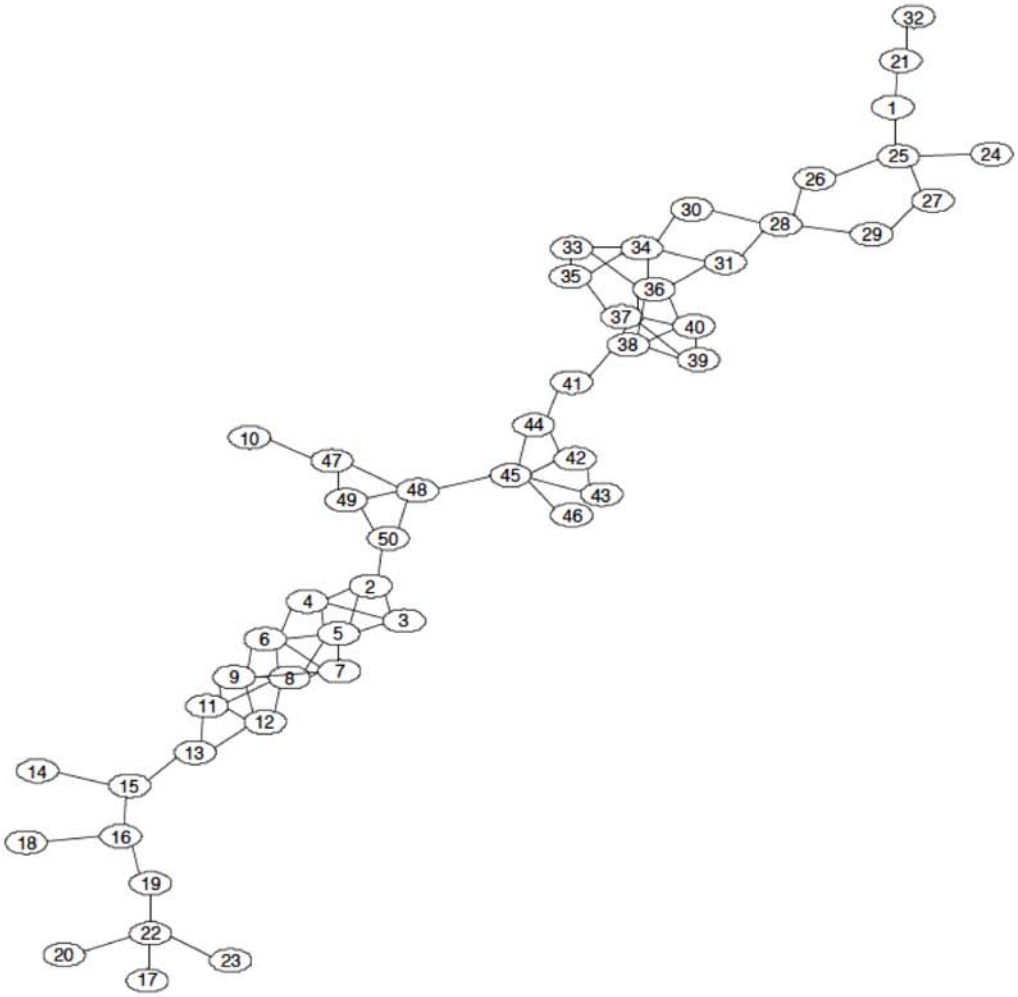


Figure 5.9. Topology of DG for Multi-Resolution Inference (EdgeNum of DG=77).

form that the approximation and detail are the first and second part of the payload respectively. When an individual mote is receiving a broadcasted belief message, it checks if the message is over a link belonging to the DG or not. If yes, the full payload will be received; otherwise only the first half of the payload (i.e., the approximation of message) will be received to save reception power. Our observation is that the multiresolution inference based on W-LBP can remove noises on those weak correlation links but keep the correlation information carried by those weak correlation links which can become useful when majority readings are missing.

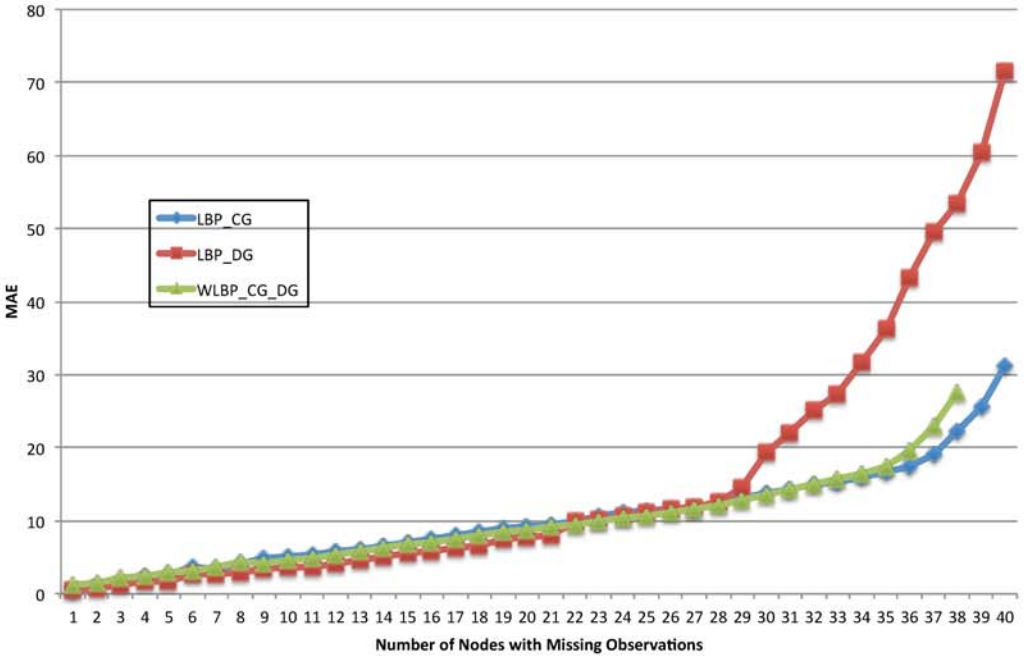


Figure 5.10. Performance comparison (IntelLab: EdgeNum of DG=77).

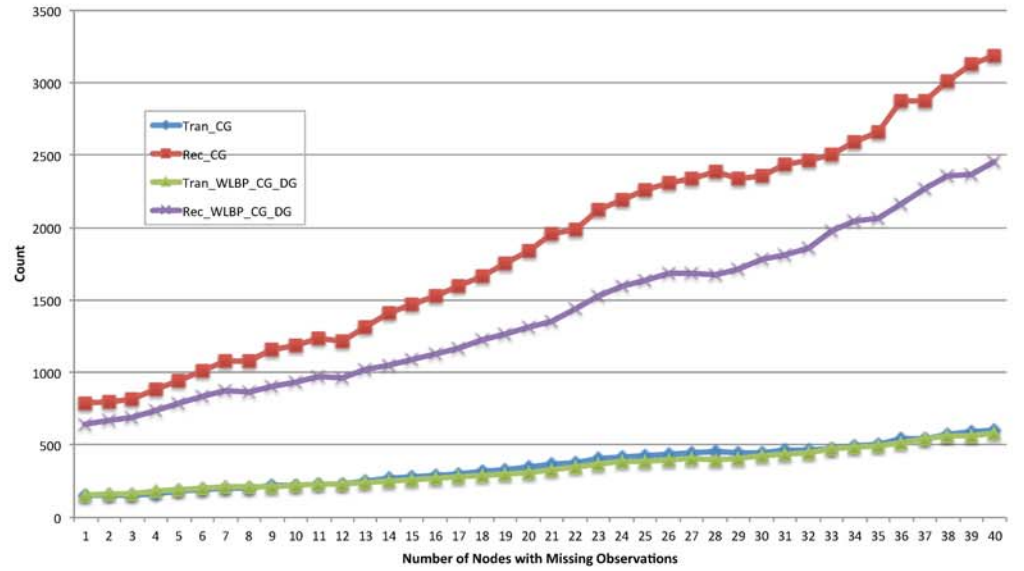


Figure 5.11. Energy efficiency comparison to CG.

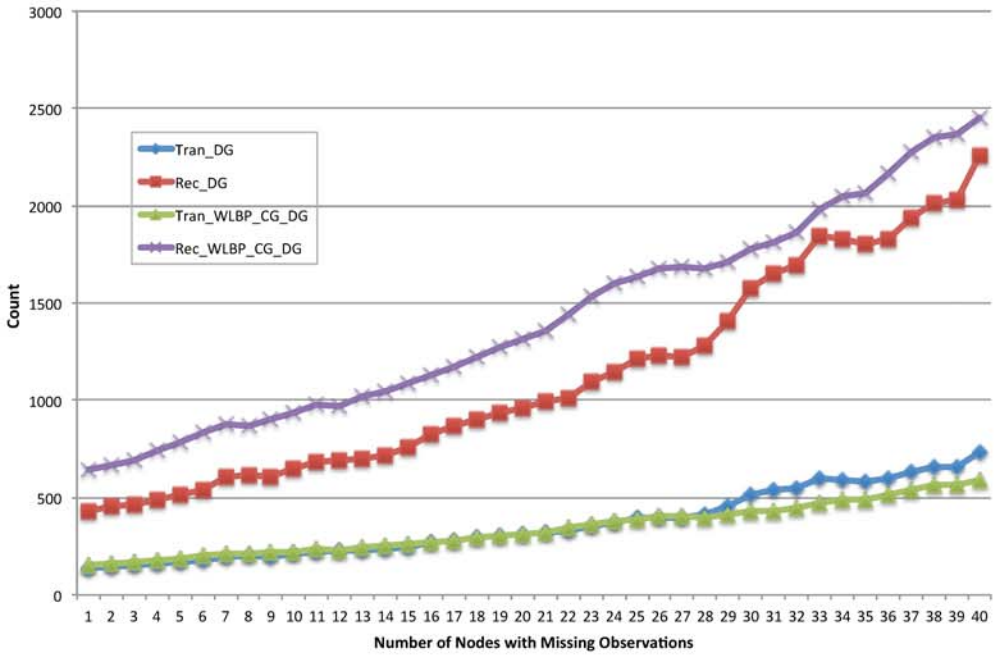


Figure 5.12. Energy efficiency comparison to DG.

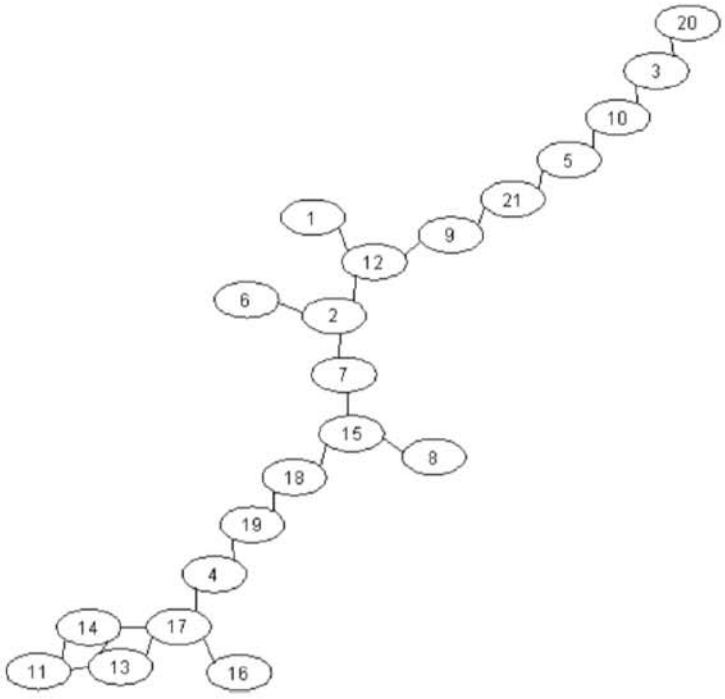


Figure 5.13. Topology of DG for Multi-Resolution Inference (EdgeNum of DG=22).

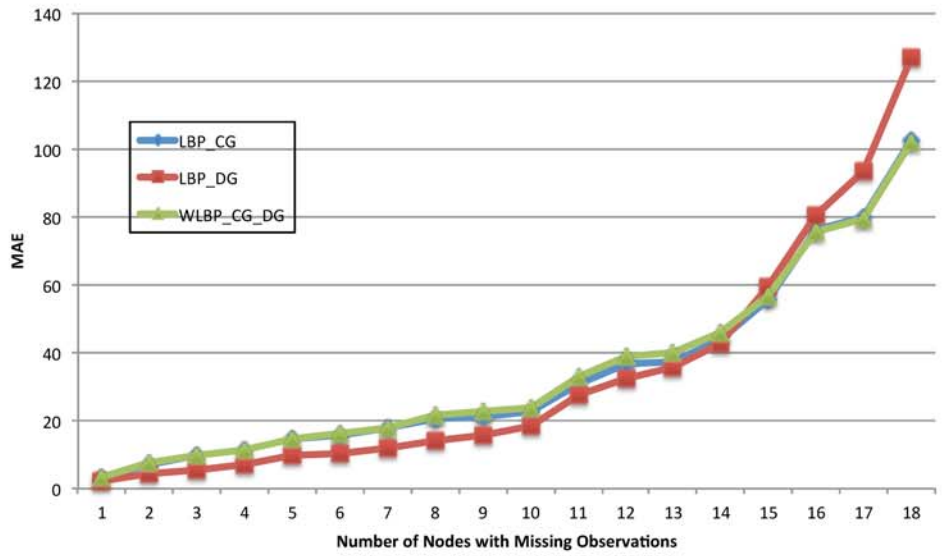


Figure 5.14. Performance comparison (Redwood: EdgeNum of DG=22).

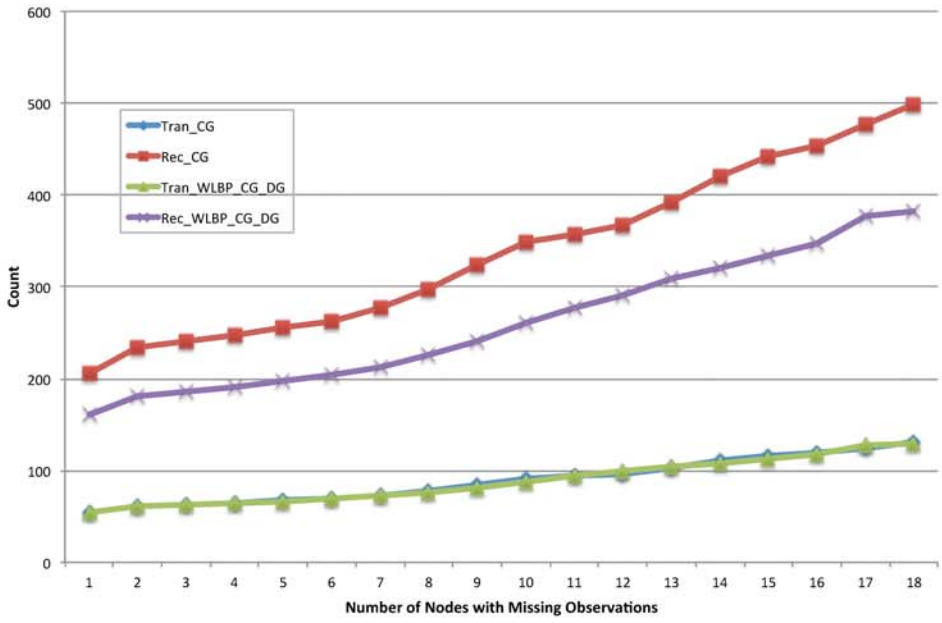


Figure 5.15. Energy efficiency analysis for outdoor sensor network.

6 SUMMARY AND FUTURE WORKS

First, we present the basic robust probabilistic estimation in a WSN to lay down the foundation of the whole set of systematic energy-efficient probabilistic estimation methods. In this initial research step we (1) present a systematic and unified MRF-based approach to estimate missing observations in WSN data gathering for real-world WSN applications; (2) show the significance of the proposed MRF framework in exploiting spatial correlations to enable energy-efficient incomplete data collections in WSNs over a long period through information inference and to support WSN fault tolerance, compared with the baseline approach, using different real-world environmental sensing data sets; (3) investigate the feasibility and efficacy of the MRF construction through automatic IPF learning with minimal training data for the task of missing information inference for environmental WSNs. Rigorous empirical study has been conducted with real-world sensor data, either outdoor and indoor, and also in sensor network of large scale: 32x32 remote sensing environmental (soil moisture and vegetation) data grids which exhibits spatial correlations in real world situation. In contrast to the baseline method of Avep procedure, with which the missing observations are estimated only using current partial observations, the proposed approach not only makes use of the current partial observations available, but also exploits the spatial correlations obtained from historical observations via MRF learning. Therefore, it is not surprising that the proposed MRF-based approach can achieve significantly higher data quality even when the unobserved nodes increase to 40% in the IntelLab sensor network. On the other hand, our work also indicates that the spatial correlation patterns learned in the MRF model of WSN is relatively stable over time comparing to Avep, since the MRF model constructed was based on the historical training data. In fact, field observations and measurements in hydrol-

ogy have provided abundant evidence that generic and repeatable spatial patterns do exist (e.g., [29]).

Although the experiment results with basic probabilistic approach are promising, the weakness is also obvious including 1) the message passing among sensor nodes is energy consuming, and the retransmission in a collision prone dense WSN makes it worse; 2) the dependence on the historical training data set to ensure an acceptable estimation accuracy. This drawback motivates the work of the learning with limited training data set, Kernel based learning and Multi-Resolution probabilistic inference in a WSN, with W-LBP and reduced Data Graph.

As illustrated in Section 5.1, based on wavelet methodology, W-LBP significantly reduces the communication volume during distributed belief inference, with very minimal degradation of estimation performance. The proposed W-LBP thus could become a better and more realistic communication basis to support distributed inference in WSNs for various applications where energy saving is crucial due to sensor nodes severe energy limitation. We demonstrate our approach through in-network estimation application using real-world sensing data. Haar wavelet was chosen due to its simplicity to implement in sensor node. Although only one level of wavelet decomposition is illustrated in our empirical study, multilevel decomposition can be adopted to achieve more substantial energy conservation. Therefore, the proposed W-LBP provides full flexibility to tradeoff inference performance with energy efficiency and opens up a new design and operational space to optimally match the specific objectives of WSNs under resource constraints. Also, due to the nature of localized communications of distributed belief inference in WSNs, the W-LBP inherits the scalability of the original LBP. We note that W-LBP is aimed for distributed belief inference at transport layer, and thus does not address MAC layer issues such as idle listening.

Following the same idea of multi-resolution inference, while W-LBP exams the possibility of more energy-efficient belief message algorithm, we devote our attention to the reasonable reduction of MRF models to better fit the requirement of energy conservation with minor performance degradation if there is any. The DG reduction

is a general data-driven approach to obtain an appropriate information structure for distributed inference in WSNs through graphical model optimization. Our approach builds a Data Graph, upon which BP-based inference is performed, from the original communication graph of WSN. The DG is constructed through two phases: graph topology structure learning and graph parameter learning. Our approach intends to construct a DG from information correlation perspective (i.e., data-driven), as a subgraph of WSN CG, with much less complexity and many fewer short loops. This way, we can achieve significant performance improvement on both in-network inference and WSN energy efficiency at the same time. As our constructed DG is a subgraph of the CG of WSN, it naturally satisfies no-routing rules for practical inference applications in WSNs. Simulation is conducted using real-world Intel-Berkeley WSNs temperature data and humidity data collected from Redwood in Sonoma, California to thoroughly evaluate our approach. Our simulation results show that based on the DG constructed by our proposed approach, the performance of distributed inference for WSN estimation application is improved compared to that based either directly on original CG or sub-CG obtained from communication perspective.

Our simulation also clearly demonstrates that in-network inference based on our constructed DG can not only generate better inference performance but also achieves significantly fewer reception/transmission operations as well as better convergence property with smaller neighborhood and fewer short loops to achieve energy savings in WSNs. We note that, when the DG is intended to support the effectiveness and efficiency of distributed inference in a WSN, the CG of the WSN is always needed to ensure the communication robustness. That is, whenever a broken link disconnects DG, an alternative message route should be found through CG. We believe such a two-level (i.e., DG and CG) topology is more applicable and flexible for effective and efficient practical use of BP-based inference in WSNs. Although the validation is conducted with real-world temperature data fusion for distributed inference in monitoring WSN, our proposed approach is general and can be applied widely to other WSN in-network inference problems with other variables or objects under consider-

ation. Based on the reduced structure, Data Graph, the idea of Multi-Resolution inference in extended further to structure field: only message over important edges (i.e. edges contained by DG) will be transmitted as original, other message will be approximated through wavelet transformation, as an effort of energy saving. The results show that this Multi-Resolution inference combined with DG shows its advantage over original inference on DG when the percentage of missing observation is high.

As a part of the systematic estimation method, it is natural to seek the energy-efficient solution in the parameter learning process after we have updated the model building and inference procedure. We present and test kernel based learning technique for parameter learning in a MRF model to greatly reduce the dependence on the historical data. This method shows its advantage when only a limited training data set is available or large training data collection is prohibited by the energy consumption. As illustrated in Chapter 3, kernel based learning procedure provides great advantage of estimation accuracy over normal IPF learning when the training data set is insufficient. Furthermore, the flexibility and robustness of kernel based learning can be clearly examined by fitting with different training sets. That is, illustrated by the experiment results, kernel based learning continuously shows advantage when the size of training data set keep increasing until the sufficient training data point is reached, in where kernel based learning has the same performance as normal learning procedure.

The proposed graphical model based inference in sensor network is still a new and active research direction and there are some interesting topics that still motivate our research for the next phrase. For kernel based parameter learning, we explored both 1D and 2D kernel to handle limited training samples. However, based on current simulation results, the performance of 2D kernel method tends to depend on the topology of the network it is applied. When 2D kernel method results better overall performance over 1D and non-kernel methods for grid network, it can not maintain the same advantage for network with irregular topology. Further research is necessary

to exam such dependence and extend the application of 2D kernel model to arbitrary topology like 1D kernel model.

For multi-resolution inference, one basic technique used is wavelet transformation that can cut the length of a belief message in half by dropping detail information when still keeps a comparable inference performance. In our simulation, one level Haar wavelet transformation was used to obtain the balance between performance and energy efficiency. If energy saving is the focus of a design for an inference application, higher level wavelet transformation can always be employed to further reduce the size of a belief message to an bearable level of information loss. The same message decomposition and reconstruction process discussed in 5.1 can be naturally extended to handle different wavelet functions and higher level wavelet transformation. The comparison analysis on choice of different wavelet functions and different levels of transformation can provide more guidance for appropriate configuration of wavelet-based belief inference for an application.

When combined with DG, we can dig deeper to the multi-resolution inference by treating edges on different DGs differently. In this thesis, the most sparse DG was implemented to achieve better energy efficiency and only most important correlation encoded in the resulted DG will be remained. For better flexibility, we can apply different levels of wavelet transformation on edges from different DGs. That is, by increasing the value of lambda, we remove more edges from DG with higher importance and we should apply higher level wavelet transformation (e.g. more than one level) on edges removed first and lower level wavelet transformation on edge removed later when remain the messages on the edges included in the most sparse DG untouched, which considered to contain most important correlation information. For clearer description, we take Figure 4.1 as an example. Assuming edges e1 to e8 were indexed by order of being removed in the process of increasing value of lambda. They will be treated equally in term of information importance and applied with one level haar wavelet decomposition in current DG based multi-resolution method as discussed in Section 4.1. With our new extension, edges e1 to e8 will be treated differently,

for instance two level wavelet transformation can be applied to e1-4 when only one level wavelet applied to e5-e8. In this way, we grant our approach a finer tuning capability that can balance the energy efficiency and performance to better meet the requirements of a real-world application. To make Multi-Resolution inference work as expected in real-world application, the MAC layer communications of WSN need to be modified accordingly to achieve this energy efficiency in multiresolution inference. For example, every mote sends its belief message in a wavelet-decomposed form that the approximation and detail are carried as the first and second half of the payload, respectively, in a belief message packet. When an individual mote is receiving a broadcasted belief message packet, it checks if the message is over a link belonging to the DG. If yes, the full payload will be received; otherwise only the first half of the payload (i.e., the approximation of message) will be received to save reception power. With those further updates from different aspects, we expect the proposed Multi-Resolution can provide better performance and adaptability to real-world applications.

REFERENCES

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *Communications Magazine, IEEE*, 40(8):102–114, 2002.

[2] T. Anker, D. Dolev, and B. Hod. Belief propagation in wireless sensor networks-a practical approach. In *Wireless Algorithms, Systems, and Applications*, pages 466–479. Springer, 2008.

[3] P. Balister, B. Bollobás, A. Sarkar, and M. Walters. Connectivity of random k-nearest-neighbour graphs. *Advances in Applied Probability*, pages 1–24, 2005.

[4] S. Bandyopadhyay, Q. Tian, and E. J. Coyle. Spatio-temporal sampling rates and energy efficiency in wireless sensor networks. *IEEE/ACM Transactions on Networking (TON)*, 13(6):1339–1352, 2005.

[5] D. Bickson, D. Dolev, and Y. Weiss. Modified belief propagation algorithm for energy saving in wireless and sensor networks. *TR-2005-85, School of Computer Science and Engineering, The Hebrew University*, 2005.

[6] Y. M. Bishop, S. E. Fienberg, and P. W. Holland. *Discrete Multivariate Analysis: Theory and Practice*. Springer, 2007.

[7] M. Cetin, L. Chen, J. W. Fisher III, A. T. Ihler, R. L. Moses, M. J. Wainwright, and A. S. Willsky. Distributed fusion in sensor networks. *Signal Processing Magazine, IEEE*, 23(4):42–55, 2006.

[8] L. Chen, M. J. Wainwright, M. Cetin, and A. S. Willsky. Multitarget-multisensor data association using the tree-reweighted max-product algorithm. In *AeroSense 2003*, pages 127–138. International Society for Optics and Photonics, 2003.

[9] L. Chen, M. J. Wainwright, M. Cetin, and A. S. Willsky. Data association based on optimization in graphical models with application to sensor networks. *Mathematical and Computer Modelling*, 43(9):1114–1135, 2006.

[10] J. Chou, D. Petrovic, and K. Ramachandran. A distributed and adaptive signal processing approach to reducing energy consumption in sensor networks. In *INFOCOM 2003, Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 2, pages 1054–1062. IEEE, 2003.

[11] T. Clouqueur, K. K. Saluja, and P. Ramanathan. Fault tolerance in collaborative sensor networks for target detection. *IEEE Transactions on Computers*, 53(3):320–333, 2004.

[12] C. Crick and A. Pfeffer. Loopy belief propagation as a basis for communication in sensor networks. In *Proceedings of the Nineteenth Conference on Uncertainty in Artificial Intelligence*, pages 159–166. Morgan Kaufmann Publishers Inc., 2002.

[13] R. Cristescu, B. Beferull-Lozano, M. Vetterli, and R. Wattenhofer. Network correlated data gathering with explicit communication: NP-completeness and algorithms. *IEEE/ACM Transactions on Networking (TON)*, 14(1):41–54, 2006.

[14] V. Delouille, R. Neelamani, and R. Baraniuk. Robust distributed estimation in sensor networks using the embedded polygons algorithm. In *Proceedings of the Third International Symposium on Information Processing in Sensor Networks*, pages 405–413. ACM, 2004.

[15] A. Deshpande, C. Guestrin, S. R. Madden, J. M. Hellerstein, and W. Hong. Model-driven data acquisition in sensor networks. In *Proceedings of the Thirtieth International Conference on Very Large Data Bases*, volume 30, pages 588–599. VLDB Endowment, 2004.

[16] A. Dogandzic and B. Zhang. Distributed estimation and detection for sensor networks using hidden markov random field models. *IEEE Transactions on Signal Processing*, 54(8):3200–3215, 2006.

[17] D. Estrin, D. Culler, K. Pister, and G. Sukhatme. Connecting the physical world with pervasive networks. *Pervasive Computing, IEEE*, 1(1):59–69, 2002.

[18] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient belief propagation for early vision. *International Journal of Computer Vision*, 70(1):41–54, 2006.

[19] S. E. Fienberg. An iterative procedure for estimation in contingency tables. *The Annals of Mathematical Statistics*, 41(3):907–917, 1970.

[20] S. E. Fienberg. *The Analysis of Cross-classified Categorical Data*. Springer, 2007.

[21] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *International Journal of Computer Vision*, 40(1):25–47, 2000.

[22] B. J. Frey and N. Jojic. A comparison of algorithms for inference and learning in probabilistic graphical models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(9):1392–1416, 2005.

[23] M. Gastpar and M. Vetterli. Power, spatio-temporal bandwidth, and distortion in large sensor networks. *IEEE Journal on Selected Areas in Communications*, 23(4):745–754, 2005.

[24] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (6):721–741, 1984.

[25] I. J. Good. *The estimation of probabilities: An essay on modern Bayesian methods*, volume 30. MIT press Cambridge, MA, 1965.

[26] I. J. Good. A Bayesian significance test for multinomial distributions. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 399–431, 1967.

[27] L. Goodman, J. Magidson, and J. Davis. *Analyzing Qualitative/Categorical Data: Log-linear Models and Latent Structure Analysis*. Abt Books, 1984.

[28] L. A. Goodman. A modified multiple regression approach to the analysis of dichotomous variables. *American Sociological Review*, pages 28–46, 1972.

[29] R. B. Grayson, G. Blöschl, A. W. Western, and T. A. McMahon. Advances in the use of observed spatial patterns of catchment hydrological response. *Advances in Water Resources*, 25(8):1313–1334, 2002.

[30] P. Hall and D. Titterington. On smoothing sparse multinomial data. *Australian Journal of Statistics*, 29(1):19–37, 1987.

[31] J. K. Hart and K. Martinez. Environmental sensor networks: A revolution in the earth system science? *Earth-Science Reviews*, 78(3):177–191, 2006.

[32] G. Hartl and B. Li. Infer: A Bayesian inference approach towards energy efficient data collection in dense sensor networks. In *Proceedings of the Twenty-Fifth IEEE International Conference on Distributed Computing Systems*, pages 371–380. IEEE, 2005.

[33] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the Thirty-Third Annual Hawaii International Conference on System Sciences*. IEEE, 2000.

[34] F. Huang and Y. Liang. Towards energy optimization in environmental wireless sensor networks for lossless and reliable data gathering. In *IEEE International Conference on Mobile Adhoc and Sensor Systems*, pages 1–6. IEEE, 2007.

[35] A. T. Ihler, J. W. Fisher, R. L. Moses, and A. S. Willsky. Nonparametric belief propagation for sensor self-calibration. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages 861–864. IEEE, 2004.

[36] A. T. Ihler, J. W. Fisher III, R. L. Moses, and A. S. Willsky. Nonparametric belief propagation for self-localization of sensor networks. *IEEE Journal on Selected Areas in Communications*, 23(4):809–819, 2005.

[37] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In *Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking*, pages 56–67. ACM, 2000.

[38] T. J. Jackson, D. M. Le Vine, A. Y. Hsu, A. Oldak, P. J. Starks, C. T. Swift, J. D. Isham, and M. Haken. Soil moisture mapping at regional scales using microwave radiometry: The southern great plains hydrology experiment. *IEEE Transactions on Geoscience and Remote Sensing*, 37(5):2136–2151, 1999.

[39] R. Jiroušek and S. Přeučil. On the effective implementation of the iterative proportional fitting procedure. *Computational Statistics & Data Analysis*, 19(2):177–189, 1995.

[40] A. Kushki, K. N. Plataniotis, and A. N. Venetsanopoulos. Kernel-based positioning in wireless local area networks. *IEEE Transactions on Mobile Computing*, 6(6):689–705, 2007.

[41] S. Z. Li and S. Singh. *Markov Random Field Modeling in Image Analysis*, volume 26. Springer, 2009.

[42] X. Luo, M. Dong, and Y. Huang. On distributed fault-tolerant detection in wireless sensor networks. *IEEE Transactions on Computers*, 55(1):58–70, 2006.

[43] S. G. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, 1989.

[44] R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng. Turbo decoding as an instance of Pearl's "belief propagation" algorithm. *IEEE Journal on Selected Areas in Communications*, 16(2):140–152, 1998.

[45] N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, pages 1436–1462, 2006.

[46] K. P. Murphy, Y. Weiss, and M. I. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 467–475. Morgan Kaufmann Publishers Inc., 1999.

[47] C. Pandana and K. R. Liu. Near-optimal reinforcement learning framework for energy-aware sensor communications. *IEEE Journal on Selected Areas in Communications*, 23(4):788–797, 2005.

[48] L. M. Parada and X. Liang. Optimal multiscale kalman filter for assimilation of near-surface soil moisture into land surface models. *Journal of Geophysical Research: Atmospheres*, 109(24), 2004.

[49] J. Pearl. *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Morgan Kaufmann, 1988.

[50] M. Penrose. *Random Geometric Graphs*, volume 5. Oxford University Press, 2003.

[51] G. J. Pottie and W. J. Kaiser. Wireless integrated network sensors. *Communications of the ACM*, 43(5):51–58, 2000.

[52] P. Ravikumar, M. J. Wainwright, J. D. Lafferty, et al. High-dimensional Ising model selection using L1-regularized logistic regression. *The Annals of Statistics*, 38(3):1287–1319, 2010.

[53] T. Roos, P. Myllymäki, H. Tirri, P. Misikangas, and J. Sievänen. A probabilistic approach to WLAN user location estimation. *International Journal of Wireless Information Networks*, 9(3):155–164, 2002.

[54] M. G. Ross and L. P. Kaelbling. Learning static object segmentation from motion segmentation. In *Proceedings of the National Conference on Artificial Intelligences*, volume 20, page 956. MIT Press, 2005.

[55] J. Schiff, J. Schiff, D. Antonelli, A. G. Dimakis, D. Chu, and M. J. Wainwright. Robust message-passing for statistical inference in sensor networks. *In Proceedings of the Sixth International Conference on Information in Sensor Network (IPSN'07)*, 2007.

[56] M. W. Schmidt, K. P. Murphy, G. Fung, and R. Rosales. Structure learning in random fields for heart motion abnormality detection. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 2, 2008.

[57] D. W. Scott. *Multivariate Density Estimation: Theory, Practice, and Visualization*, volume 383. John Wiley &Amp, 2009.

[58] K. Sha and W. Shi. Consistency-driven data quality management of networked sensor systems. *Journal of Parallel and Distributed Computing*, 68(9):1207–1221, 2008.

[59] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*, volume 26. CRC press, 1986.

[60] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for markov random fields. In *Computer Vision–ECCV 2006*, pages 16–29. Springer, 2006.

[61] G. Tolle, J. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, P. Buonadonna, D. Gay, et al. A macroscope in the redwoods. In *Proceedings of the Third International Conference on Embedded Networked Sensor Systems*, pages 51–63. ACM, 2005.

[62] M. Vetterli and C. Herley. Wavelets and filter banks: Theory and design. *IEEE Transactions on Signal Processing*, 40(9):2207–2232, 1992.

[63] M. C. Vuran, Ö. B. Akan, and I. F. Akyildiz. Spatio-temporal correlation: Theory and applications for wireless sensor networks. *Computer Networks*, 45(3):245–259, 2004.

[64] M. C. Vuran and I. F. Akyildiz. Spatial correlation-based collaborative medium access control in wireless sensor networks. *IEEE/ACM Transactions on Networking*, 14(2):316–329, 2006.

[65] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. Tree-based reparameterization framework for analysis of sum-product and related algorithms. *IEEE Transactions on Information Theory*, 49(5):1120–1146, 2003.

[66] M.-C. Wang and J. Van Ryzin. A class of smooth estimators for discrete distributions. *Biometrika*, 68(1):301–309, 1981.

[67] C. J. Willmott and K. Matsuura. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research*, 30(1):79, 2005.

[68] F. Xue and P. R. Kumar. The number of neighbors needed for connectivity of wireless networks. *Wireless Networks*, 10(2):169–181, 2004.

[69] W. Ye, J. Heidemann, and D. Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Transactions on Networking*, 12(3):493–506, 2004.

[70] J. S. Yedidia, W. T. Freeman, Y. Weiss, et al. Generalized belief propagation. In *Neural Information Processing Systems (NIPS)*, volume 13, pages 689–695, 2000.

[71] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

[72] W. Zhao and Y. Liang. A systematic probabilistic approach for estimation in dense wireless sensor networks. In *Wireless Communications and Networking Conference, 2008*, pages 3285–3290. IEEE, 2008.

[73] W. Zhao and Y. Liang. W-LBP: Wavelet-based loopy belief propagation for wireless sensor networks. In *Third International Conference on Sensor Technologies and Applications*, pages 617–622. IEEE, 2009.

[74] W. Zhao and Y. Liang. A systematic probabilistic approach to energy-efficient and robust data collections in wireless sensor networks. *International Journal of Sensor Networks*, 7(3):162–175, 2010.

[75] W. Zhao and Y. Liang. Kernel-based markov random fields learning for wireless sensor networks. In *IEEE Thirty-Sixth Conference on Local Computer Networks (LCN)*, pages 155–158. IEEE, 2011.

[76] W. Zhao and Y. Liang. Inference in wireless sensor networks based on information structure optimization. In *Proceedings of IEEE Thirty-Seventh Conference on Local Computer Networks (LCN)*, pages 551–558, 2012.

VITA

VITA

Wei Zhao received the B.S. degree in automation control and the M.S. degree in artificial intelligence from Harbin Engineering University, Heilongjiang, China, in 2001 and 2004 respectively. He was a research assistant with Virginia Polytechnic Institute and State University, Blacksburg, Virginia from 2005 to 2007. From 2008 to 2013, he was a research assistant at the Purdue University-Indiana University, Indianapolis, Indiana. His research interests include data fusion and inference in wireless sensor network, graphical model, machine learning and statistics for data estimation.